WELFARE DATA TRACKING IMPLEMENTATION PROJECT

INTEGRATION/SYSTEM TEST SIGN-OFF

APPENDIX I
   A.   SCENARIO EXECUTIONS REPORT
   B.   SCRIPT-INCIDENT CROSS REFERENCE TABLE
   C.   FINAL TESTING STATUS REPORT

APPENDIX II   **BUSINESS REQUIREMENTS AND SCENARIO SCRIPT ASSOCIATIONS**

APPENDIX III   **UAT SCRIPTS**

APPENDIX IV   **UAT REQUEST FOR PARTICIPANTS LETTER**

APPENDIX V   **UAT TEAM ROSTER**

APPENDIX VI   **INCIDENT-ISSUE IDENTIFICATION FORM**

APPENDIX VII   **UAT SCRIPT COMPLETION FORM**

APPENDIX VIII   **UAT SCRIPT TESTING SCHEDULE**

APPENDIX IX   **SAMPLE DAILY UAT STATUS REPORT**

APPENDIX X   **SAMPLE DAILY UAT INCIDENT STATUS REPORT**

APPENDIX XI   **SAMPLE DAILY UAT PCR STATUS REPORT**

APPENDIX XII   **REGRESSION TESTING SCENARIOS**

APPENDIX XIII   **UAT ACCEPTANCE CRITERIA**

1. Project Overview

1.1   Document Purpose

The purpose of the Welfare Data Tracking Implementation Project (WDTIP) **Integration/System Test Sign-off** deliverable is to provide a comprehensive description of the findings identified during integration/system testing (referred to hereafter as "system test"). It provides a detailed description of the system test procedures as well as the planned User Acceptance Test (UAT) activities of the WDTIP system. This deliverable will address both incidents and issues that arose during the course of system test scenario execution. The WDTIP UAT Plan section of this document details the Project's approach to the planning and execution of all WDTIP system UAT activities.

1.2   Project Overview

This section provides an overview of the WDTIP, delineating project definition, purpose, objectives and scope to provide the reader with the context for decisions made regarding the system and user acceptance testing planning and associated activities.

1.2.1   Project Definition

The WDTIP is a system development project that includes overall project management; designing, building and testing the system; developing and executing user training; communicating with internal and external stakeholders; and deploying the system. In addition, data will be converted from county systems to the WDTIP database. It is anticipated that this data conversion will entail both automated and manual methods. Subsequent ongoing batch data loads from the counties are also included in the WDTIP. WDTIP scope is detailed in the *1.2.2 Project Scope* subsection below.

In response to the Personal Responsibility and Work Opportunity Reconciliation Act (PRWORA) of 1996, the State of California passed Assembly Bill (AB) 1542. AB-1542 institutes the Temporary Assistance to Needy Families (TANF) program in California and imposes welfare time limits, as well as new programmatic and eligibility rules. In addition to welfare time limits, AB-1542 mandates work requirements through the California Work Opportunity and Responsibility to Kids (CalWORKs) program. As a result of the CalWORKs program, county welfare departments are required to have a mechanism to track eligibility time limits, and other related data on an individual level, across counties and over time to comply with the tracking requirements of both State and Federal mandates.

The purpose of the WDTIP is to provide a communication mechanism and central data repository that can be accessed by all technology-enabled counties and relevant agency systems to meet the requirements of TANF and CalWORKs legislation. WDTIP addresses the immediate need for Federal and State Welfare Reform tracking functions imposed by the Federal PRWORA, AB-1542 and relevant All County Letters (ACLs) issued by the California Department of Social Services (a list of these ACLs is included as *Appendix 1-1* in the **Implementation Plan** deliverable).

The WDTIP objectives are to satisfy the aforementioned legislative requirements by providing an automated statewide repository for Welfare Reform data elements and to facilitate communication between disparate county welfare and statewide welfare-related systems. The primary data to be collected, calculated (if necessary), and tracked for applicants/recipients includes:

- ❑ TANF 60-month clock
- ❑ CalWORKs 60-month clock
- ❑ Welfare to Work (WTW)18/24-month clock

1.2.2 Project Scope

The overall objective of the WDTIP is to provide a communication mechanism and central data repository that can be accessed by all technology-enabled counties and relevant agency systems. In addition, the WDTIP system must enable counties to meet the requirements of Welfare Reform. The scope of the WDTIP includes design, construction, testing and implementation of the WDTIP system. This system will enable all 58 California counties to accurately track individual welfare recipient information to meet the requirements of both State and Federal Welfare Reform. WDTIP is also tasked with development of Customer Information Control System (CICS) screens that will provide counties with the ability to view data, perform inquiry and online updates and create management reports.

In addition to the WDTIP system development tasks listed above, the WDTIP is responsible for conducting a one-time data conversion of county data. This one-time conversion will be required for the initial county data load into the WDTIP database. The counties will perform subsequent ongoing data loads. Examples of data to be tracked in the WDTIP system include:

- ❑ PRWORA time clock calculation
- ❑ CalWORKs time clock calculations, including exceptions and exemptions
- ❑ Diversion program and payment information
- ❑ Sanction information to provide appropriate CalWORKs sanction data across counties

The data conversion of county data to populate the WDTIP database will be a vital component of the WDTIP. Some counties have not maintained the level of historical data necessary to provide the initial county-specific information required for optimal time clock calculations. In addition to this, because the Medi-Cal Eligibility Data System (MEDS) does not supply all the needed data, SAWS Information System (SIS) cannot currently provide complete data tracking or correctly calculate cumulative time-on-aid. To calculate accurate time-clock data, the WDTIP database must be populated with direct county data via a one-time data conversion and ongoing updates.

The WDTIP Implementation and WDTIP Application Teams (with input from the counties) developed conversion specifications and standard file formats that will support

each county's conversion and update efforts. Each county will now be responsible for providing the conversion data files to populate the WDTIP database. It is expected that counties or their consortium will provide WDTIP with ongoing data files. County technical resources will be needed to produce the conversion extract. It is important that all counties participate in this conversion effort for the WDTIP system to generate complete, accurate and meaningful data.

It is expected that the quality of the county data and the resulting time clock calculation will improve as counties begin using the WDTIP system to collect and maintain the required time tracking data elements. The scope of WDTIP includes assisting the counties with the following data conversion activities: design, development, testing and implementation of conversion programs including, but not limited to, the following:

- Identification of required county data elements to populate the WDTIP database (completed)
- Identification of county file format requirements (completed)
- Development of edit and error processing rules (completed)
- Development of ongoing load requirements for county data into the WDTIP database (completed)
- Assistance with the one-time initial data conversion
- Assistance for the county technical resources when developing the data extract requirements

In addition, the WDTIP is responsible for the following implementation activities:

- Regional information sessions (completed)
- Regional training sessions
- County visits as needed
- Consistent communication with stakeholders
- Implementation support

The scope of the WDTIP does not include:

- Providing counties with resources to convert county data into the standard file (for initial data conversion and ongoing data loads)
- Assisting agencies/counties with the design and development of county-specific welfare screens to view WDTIP data
- Developing or managing changes to the Statewide Client Index (SCI) application

2. Acronyms and Definitions

The table below provides a list of commonly used acronyms within this document.

Table 2-1: Acronyms

Acronym	Phrase/Name
AB	Assembly Bill
ACL	All County Letter
AV	Audio Visual
CalWORKs	California Work Opportunity and Responsibility to Kids
CDHS	California Department of Health Services
CDSS	California Department of Social Services
CICS	Customer Information Control System
CIN	Client Identification Number
CWDA	California Welfare Directors Association
HHSDC	California Health and Human Services Agency Data Center
ISAWS	Interim Statewide Automated Welfare System
JRP	Joint Requirements Planning
LEADER	Los Angeles Eligibility, Automated Determination, Evaluation and Reporting System
MEDS	Medi-Cal Eligibility Data System
PC	Personal Computer
PCR	Program Change Request
PRWORA	Personal Responsibility and Work Opportunity Reconciliation Act
PTS	Project Tracking System
SAWS	Statewide Automated Welfare System
SCI	Statewide Client Index
SIS	SAWS Information System
SME	Subject Matter Experts
SSN	Social Security Number

Acronym	Phrase/Name
STS	Scenario Tracking System
TANF	Temporary Assistance for Needy Families
TRAC	Tracking Recipients Across California
UAT	User Acceptance Test
WAN	Wide Area Network
WCDS	Welfare Case Data System
WDTIP	Welfare Data Tracking Implementation Project
WTW	Welfare to Work

The table below provides definitions of potentially unfamiliar terms and phrases used throughout this document.

Table 2-2: Definitions

Term or Phrase	Definition
Acceptance Criteria (System Test)	Criteria identified by the WDTIP Project Management Team outlining acceptable WDTIP system testing results.
Acceptance Criteria (UAT)	Criteria identified by CDSS outlining acceptable UAT results.
Backup Data	Additional data created for UAT in the event the original data becomes corrupt or lost.
Batch	An automatic process that happens within the WDTIP system on a predetermined schedule.
CalWORKs 60-Month Time Clock	A calculation of the maximum period of time an individual is allowed to receive benefits under the State CalWORKs Program. This clock began to tick in January 1998.
Clock Roll	A particular time clock advancement designed to coincide with a testing script or step.
Conversion	The point in time after which a county sends information to the WDTIP system through the extraction file process.
Conversion Load	The initial information sent to WDTIP database by an individual county/consortia and entered into WDTIP electronically.
County/Consortia	Refers to the eight core systems that will be used for the initial direct county data loads as well as the ongoing data loads. Representatives from these counties/consortia have been solicited for assistance throughout the duration of the WDTIP Project, including UAT Team participation.

Term or Phrase	Definition
Cycle	The performance of the complete set of testing scenarios.
Data Build	Information created to support a script at a particular point in time.
Data Load	The process of entering data into the WDTIP system using a daily or monthly batch.
Data Set	Listing of 150 Client Identification Numbers (CIN) with specific program and demographic information attached.
Data Sources	The systems that will be sources of data for the initial and ongoing data loads of the WDTIP database.
Enhancement	Any incident that is outside the Project's scope. An enhancement is subject to the established protocol of the Configuration Control Process.
Exception File	Electronic file created when data sent from a county/consortia cannot be processed by the WDTIP system either due to a problem with the entire file or a problem with a single record. This file is sent on a daily basis to the system of origin so that data can be corrected and resent.
Fail (System Test)	The results of testing a scenario, script or step that does not meet all of the expected results.
Fail (UAT)	The status of a script that has not been signed-off by one or more members of the UAT Team because actual results do not meet expected results.
Federal/State Time Clocks	The time clocks that will be calculated and tracked in the WDTIP system: the TANF 60-month time clock, the CalWORKs 60-month time clock and the WTW 18/24-month time clock.
File Extract Load	Information entered into WDTIP via electronic file.
Fix	Any incident that is within the Project's scope. Fixes have been documented in the appropriate tracking system and assigned to the appropriate programmer to make the necessary adjustments to the affected system programs.
Focus Area	How incidents are categorized at subsystem level (online, batch processing, reporting).
Incident	An incident occurs when the actual result of one or more steps in a script differs from the script's expected results or when system functionality differs from user expectations.
Initial Data Load	The initial population of the WDTIP database with historical SIS, MEDS, and direct county data.
Inquiry	The ability to view information in the WDTIP system without the ability to update it.
Integration Testing	System Testing. Performance of scenarios that test the WDTIP system functionality and processing.

Term or Phrase	Definition
Log-on	A password protected identification used to access the WDTIP system.
Mandatory Data Elements	The 42 identified data elements that are required for accurate calculation of the Federal/State time clocks.
Online	The WDTIP system screens where welfare-related information is displayed for the user.
Pass (System Test)	The results of testing a scenario, script or step that has successfully met the expected results.
Pass (UAT)	The status of a script that has been tested by each member of the UAT Team and all members have found that actual results meet expected results.
Pending (UAT)	A script that has not been classified as pass or fail because the Team is waiting for an action to occur before the script can be fully tested (e.g., the Team is awaiting specific MEDS logons).
Pending (System Test)	A script that meets the passing criteria but has a pending policy clarification required or enhancement requested.
Priority, High	A designation given to a script that should be completed during UAT testing. The script will focus on a general area or business requirement. The script will be dependent on a designated clock roll.
Priority, Low	A designation given to a script that should be completed during UAT testing. The script will focus on a specific area or business requirement. The script will not be dependent on a designated clock roll.
Priority, Medium	A designation given to a script that should be completed during UAT testing. The script will focus on a specific area or business requirement. The script may be dependent on a designated clock roll.
Production Environment	The area in WDTIP that is accessed by actual users and contains actual participant information.
Program Change Request	A request for a programming change assigned to a specific developer when a programming problem is encountered in the system and tracked through the Project Tracking System.
Project Tracking System (PTS)	A system written in Microsoft Access to track WDTIP system incidents, issues and change requests.
Regression Testing	Testing to ensure that a correction to the system has been successful.
Scenario	A set of scripts used to test a focus area of the WDTIP system.

Term or Phrase	Definition
Scenario Tracking System (STS)	A system developed in Microsoft Access which tracks the test dates and test results of scenarios, scripts and steps.
Script	A scenario component consisting of multiple steps designed to test a specific area.
Script Sign-off	Formal written acceptance by the UAT tester of the WDTIP functionality contained within the tested script.
Snapshots	Periodic electronic copying of data at a point in time.
Step	A specific procedure within a script.
Subject Matter Expert (SME)	An individual with an intimate knowledge of welfare regulations and procedures.
System	The WDTIP system.
System Test	Performing scenarios which test all WDTIP system functionality and processing.
TANF 60-Month Time Clock	A calculation of the maximum period of time an individual is allowed to receive benefits under the Federal TANF Program. This clock began to tick in December 1996.
Test Environment	Separate area within the WDTIP system used specifically for test purposes. This area does not affect production data.
Time Clock Advancement	Adjusting the date within the WDTIP system to simulate a future point in time.
UAT Incident Identification Form	A form that will be used by the UAT Team to record any incident that is discovered during the testing process.
UAT Team	A group of 10 individuals representing the State and some counties for the purpose of UAT testing activities.
UAT Team Lead	WDTIP State staff member who will lead the UAT activities.
Unit Testing	Tests each WDTIP module separately.
Update	The ability to enter information into the WDTIP system that will be stored and used to calculate time clocks.
User Acceptance Test (UAT)	The process by which representative users test the system screens, navigation and functionality.
User Generated Data	Information entered into WDTIP manually by the user.
WDTIP Database	The database which will support the WDTIP system and be populated with historical SIS, MEDS, and direct county data.
WDTIP System	The system that is being developed to accept direct county data, calculate the Federal and State time clocks and display the resulting information.

Term or Phrase	Definition
WTW 18/24-Month Time Clock	A calculation of the maximum period of time an individual is allowed to receive benefits under the State WTW Program. This clock begins to tick when the individual recipient signs or refuses to sign a WTW Plan.

3. Integration/System Test Sign-off Introduction

3.1 Document Objective

The **Integration/System Test Sign-off** deliverable provides a comprehensive description of the findings identified during system testing of the WDTIP system as well as a detailed description of UAT activities. The document contains the procedures followed during system test and addresses both incidents and issues that arose during the course of scenario execution. The WDTIP UAT Plan section of this document details the Project's approach to the planning and execution of all WDTIP system UAT activities.

3.2 Document Purpose and Audience

The purpose of the **Integration/System Test Sign-off** is to document findings compiled during system test and provide the definition and objective of UAT activities. This document includes assumptions and constraints used in the development of the strategy and approach. UAT activities refer to those activities that must be completed in order for the California Department of Social Services (CDSS) and county and consortia representatives to validate and sign-off on the WDTIP system functionality (as it was designed) and ease of use as they pertain to the user's business needs.

This document provides a comprehensive description of our approach to UAT, targeting non-technical individuals. It also provides the detailed acceptance criteria necessary for the UAT Team to successfully complete UAT.

3.3 Document Scope

The **Integration/System Test Sign-off** document includes the following information:

- ❑ *Project Overview* – This section contains a brief description of the purpose of this document as well as an overview of the Project and the scope.

- ❑ *Acronyms and Definitions* – This section provides two reference tables, a list of commonly used acronyms and a list of definitions for potentially unfamiliar terms and phrases used throughout the document.

- ❑ *Integration/System Test Sign-off Introduction* - This section contains the objective, purpose and scope of the **Integration/System Test Sign-off** deliverable.

- ❑ *System Test Results* – This section contains an overview of the procedures followed during system test and the test results. This section includes the definition and objective of system test and outlines acceptance criteria used to define the successful completion of system test. In addition, this section includes the approach (tools used, schedule and focus areas defined) and the process (how scripts are executed, how incidents are tracked, etc.). Lastly, this section outlines the WDTIP system test findings and conclusions. The findings include information on issues that were identified during the course of system testing.

- ❑ *User Acceptance Test Plan* - This section provides the definition and objective of UAT and includes assumptions and constraints as well as the strategy and approach used in developing the UAT Plan. This section also includes detailed information

regarding the approach to UAT scenario and script development, recruitment of UAT Team members, UAT facility requirements and procurement, system testing region development and maintenance, data generation, UAT acceptance criteria, testing and retesting processes, and daily and summary testing requirements. The UAT scripts are included as *Appendix III* of this document.

4   System Test Results

4.1   Definition

For the purposes of WDTIP, system test is defined as validating (through scenario and script execution) that the TRAC application meets all of the pre-defined WDTIP business and technical requirements and functions as designed.

Accordingly, the purpose of system testing was to exercise all aspects of the TRAC application in preparation for User Acceptance Test activities and to validate that all WDTIP system requirements were successfully met in the application. System test was planned and executed to ensure that the testing:

❑   Stretched across all functional areas and business requirements of the TRAC application

❑   Validated that all TRAC application programs, interfaces and procedures operate together as designed and intended

❑   Was process-related

❑   Was formalized with scripted test cases (scenarios) that document expected results

❑   Was performed by an objective, cross-functional team who was not involved with the system design

❑   Documented failures and re-tests after corrections had been made

4.2   Objective

The primary objective of the WDTIP system test was to thoroughly test the TRAC application in a controlled and structured manner to validate that the technical and business requirements outlined in the **Updated Business Requirements** deliverable had been implemented. This objective was accomplished by:

❑   Testing the online functionality of each TRAC subsystem and its interaction with all other TRAC subsystems. Each item tested was considered complete when its associated acceptance criteria had been met and documented. The results of this testing were documented in the Scenario Tracking System

❑   Testing the batch functionality of all TRAC batch programs and their interaction with all components of the TRAC application. This testing included validating the accuracy of batch procedures and job control language. The results of this testing were documented in the STS. Each tested item was considered complete when its associated acceptance criteria had been met and documented

❑   Validating that all WDTIP system requirements were met

❑   Validating the correctness of the TRAC application and equipment configuration on each tier, as it relates to the TRAC system test environment

❑   Verifying the functionality and operation of the TRAC application (online and batch) using simulated "converted" data

❑ Validating and maintaining testing and system documentation throughout the testing process

❑ Performing quality assurance testing

❑ Establishing and managing the system test environment and resources

❑ Culminating system test activities with WDTIP Project Management approval to enter UAT

The remainder of *Section 4, System Test Results*:

❑ Outlines the WDTIP system test acceptance criteria

❑ Identifies the approach and key processes used for system testing

❑ Summarizes the system test findings

4.3   Acceptance Criteria

The system test acceptance criteria outline acceptable system testing results and were determined by WDTIP Project Management. The criteria state that all steps must "pass" at the conclusion of system testing. This means that all scripts and scenarios must also "pass", since scenarios are comprised of scripts, and each script has multiple steps. (For details on scenarios, scripts and steps, please reference *Section 4.4.1.1, Scenarios and Scripts*).

Outcomes of testing the steps could include one of three possible end results: pass, fail, or pending. Steps classified as having "passed" are those where the actual results in the system matched the expected results detailed in the scripts. Steps classified as having "failed" are those where the actual results did not meet the expected results. "Pending" steps were those where the expected results met the actual results but a policy clarification or additional information was required from an entity outside the WDTIP. All three results were tracked and monitored in the WDTIP STS and, as appropriate, in the Project Tracking System (PTS).

Steps (and corresponding scripts and scenarios) which failed, followed the procedures outlined in *Section 3.1.2, Fix Request Procedures, Appendix 2* of the **Completed Source Modules/Unit Test** deliverable. Steps that were classified as pending were reviewed with the Project Management Team and were resolved on a case-by-case basis.

4.4   Approach

This section describes the approach used to execute system testing. The following subsections discuss the tools used, outline the system test schedule, and describe the approach to testing the four focus areas: online, batch processing, reporting and database.

4.4.1   Tools
The primary tools used for system testing were the test scenarios and scripts. Other tools used during system testing were the PTS and the STS. These systems tracked issue,

incident and scenario status. Each is described in more detail in the paragraphs that follow.

4.4.1.1 Scenarios and Scripts

Scenarios focus on specific functional areas of the system and are comprised of individual scripts. Scripts provide detailed information about the situation to be tested as well as step-by-step instructions to guide a tester through the script. Scripts also provide the expected results of each step taken so that the tester can verify that the system functions and displays information correctly.

The system test scenarios were designed to ensure effective test coverage of all requirements of the WDTIP system. The System Test Team reviewed specifications for each program before preparing the system test scenarios. Additionally, each TRAC program was traced to the **Updated Business Requirements** deliverable to identify the corresponding program that would provide that functionality. This cross-reference is included in *Section 3, Traceability Matrix,* of the **Completed Source Code Modules/Unit Test** deliverable and demonstrates that the required WDTIP functionality was incorporated into the testing plan.

The System Test Team identified 13 scenarios addressing broad functional areas of the system. One or more scripts (for a total of 178) were created for each scenario and correspond to a specific focus area, including online screens, reports, batch processing and database. Each script was composed of a series of steps, each having its own expected result. The steps provided detailed instructions on what input was required to achieve the expected result (output). Please refer to the STS for the detailed scenarios, scripts and steps used during system test.

The table below documents the identified scenarios and the number of scripts contained in each scenario.

Table 4-1: Scenario Distribution

Scenario Number	Scenario Name	Number of Scripts*	Focus Area
1	Validate Screen Navigation – Update Screens	4	Online
2	Validate Screen Display – Update Screens	4	Online
3	Validate Help Functions – Update Screens	13	Online
4	Functionality – Update Screens	1	Online
5	Validate Time Clocks – Update Screens	13	Online
6	Validate Screen Navigation – Inquiry Screens	17	Online
7	Validate Screen Display – Inquiry Screens	17	Online
8	Validate Help Functions – Inquiry Screens	37	Online
9	Functionality – Inquiry Screens	18	Online
10	Validate Time Clocks – Inquiry Screens	29	Online

Scenario Number	Scenario Name	Number of Scripts*	Focus Area
11	Validate Report Data	4	Reports
12	Validate Load and Time clocks	20	Batch
13	General – Database Functionality	1	Database

* This number does not represent the steps associated with each script.

4.4.1.2 Project Tracking System (PTS)

The PTS is a tool used to monitor both issues and incidents that arise during the course of the WDTIP Project. The PTS is an Access database located on the shared drive of the SAWS network. An incident is defined (in the WDTIP **Configuration Management Plan** deliverable) as a program logic anomaly identified in the TRAC Application during system and user acceptance testing. The primary purpose for incident tracking is to help monitor the progression of incidents through the Change Control Process and produce reports that provide information regarding status and assignment of incidents. Please refer to *Section 8.1.1, Configuration Issue Tracking System* of the **Configuration Management Plan** deliverable and the **Project Management Plan** deliverable for more information on issue tracking**.**

4.4.1.3 Scenario Tracking System (STS)

The STS is a tool used to monitor the status of test scenarios. The STS is an Access database located on the shared drive of the SAWS network. The STS allows for the reporting and monitoring of scenarios, scripts and steps along with associated expected results. Test results (by step) were recorded in the STS to identify status of scenarios and scripts. System test progress was reported in the daily WDTIP status meetings using reports generated by this system.

4.4.2 Schedule

Approximately two months were designated for WDTIP system testing. To maximize the effectiveness of testing during this time, it was important to execute scripts in an order that considered system and test dependencies as well as other technological requirements of the application. The order and dependencies for test script execution were detailed (including outlining items that could be tested in parallel) in *Appendix 2, System Test Workplan*, of the **Completed Source Module/Unit Test** deliverable.

The system test schedule reflected the logical order and test schedule of TRAC component (program module) testing. It was designed to first test non-data dependent functionality (i.e., screen navigation and screen display) followed by testing system functionality that did depend upon data (i.e., batch processing and time clock validation).

The testing occurred in three cycles. In the first two cycles, the System Test Team executed all scripts (online, batch, reports and database); in the third cycle, the Team executed only the batch scripts. In Cycle I, retesting the goal was to completely execute all scripts. During Cycle II, the Team executed all scripts again and retested those scripts that did not pass in Cycles I or II. Cycle III focused on full execution of all batch scripts with the goal being a "clean" end-to-end run.

System test was conducted following the schedule outlined in the table below.

Table 4-2: WDTIP System Testing Schedule

Functional Area	Cycle Number	Start Date	End Date
Online	I	January 3, 2000	January 31, 2000
Batch	I	January 17, 2000	February 4, 2000
Reports	I	January 31, 2000	February 2, 2000
Database	I	January 31, 2000	February 2, 2000
Online	II	February 3, 2000	February 22, 2000
Batch	II	February 1, 2000	February 11, 2000
Reports	II	February 23, 2000	February 25, 2000
Database	II	February 23, 2000	February 25, 2000
Batch	III	February 14, 2000	February 29, 2000

4.4.3 Focus Areas

The following paragraphs describe the approach to testing the online, batch processing, reporting, and database focus areas.

4.4.3.1 Online

Testing for online occurred in Cycles I and II. In both cycles, every script and step were executed at least twice. Data representing 2,000 unique individuals with varied program participation was tested. The objective of the first cycle was to filter out any obstacles (e.g., connection failure, incorrect data) to completing all of the scripts. The second cycle of testing sought to resolve any discrepancies identified during the first cycle of testing (between actual and expected script results), and to ensure that changes made to the programs did not affect other system functionality. The testing of the online screens examined screen navigation, screen displays, field/screen help, user authorization (including update versus inquiry access) as well as the add, modify and delete functionality of the online update screens. Additionally, time clocks were validated for changes that were made to data through the online update screens by adding, modifying and/or deleting records.

4.4.3.2 Batch Processing

Batch testing was completed during all three cycles. The first cycle focused on validating the functionality for the initial data load (from SIS), the ongoing loads from MEDS and the eight source systems, as well as the time clock calculations. The second cycle of batch testing sought to resolve any issues identified during the first cycle. The objective of the final cycle of batch testing was to complete an end-to-end test of batch processing without incident. Any discrepancies that were identified or remained outstanding during the final cycle of testing were regression tested when the incidents were resolved.

The methodology used allowed the system test to cover the entire time clock life cycle (60+ months for TANF and CalWORKs, and 18/24+ months for WTW). This was accomplished in a compressed testing timeframe. The System Test Team accelerated time by running more than one batch processing day per calendar day and advancing the processing date, as necessary. For the purposes of system test, a batch processing day is equivalent to running all batch programs (county load files and daily time clock calculation module) for a particular calendar day. The time clocks for the test individuals were advanced rapidly by running the monthly time clock calculation module for every month from February through December for each test year.

In order to accurately simulate the production environment, the system test database needed data as a "starting point" (i.e., to represent the SIS conversion into the TRAC database at go-live). Hence, test individuals were loaded into the TRAC database as part of an initial load. The set of individuals that were used for system test was created such that their program participation data represented various points in the time clock lifecycle. Some individuals had TANF program participation prior to December 1996 (the start of TANF). Additionally, some individuals had CalWORKs program participation prior to January 1998 (the start of CalWORKs). Because the earliest expiration for a CalWORKs time clock will occur on January 2003, the processing date was advanced to February 2003 to exceed both TANF and CalWORKs time clocks. The WTW 18/24-month time clock was also exceeded within the test for some individuals.

The batch testing imposed modifications to particular individuals from two perspectives:

- ❑ Transaction: the data associated with an individual changed (e.g., an individual started a new program or received a diversion payment)
- ❑ Time: as each month passed, the time clocks needed to reflect the passage of time

A combination of these two factors was used to test within the relevant batch scripts.

Each batch cycle simulated the processing of daily loads from five different counties (i.e., five unique county load files per batch processing day). Within each county load file there were 12 unique individuals, or welfare recipients, for which at least one transaction (e.g., an add, modify or delete) was included. Each cycle used a different set of individuals.

In Cycle I, the System Test Team:

- ❑ Loaded initial data to represent the SIS conversion to the TRAC database
- ❑ Executed six daily processing jobs, each job consisting of the five county load files and daily time clock calculation. The processing dates of the jobs were advanced to rapidly move the clock ahead and spanned from January 10, 2000 to January 28, 2000
- ❑ Executed a monthly time clock calculation job to signify end of month (January 2000) processing
- ❑ Executed two additional daily processing jobs (again, advancing the processing date to get to end of month) to signify transactions for the next month (February 2000)

- ❑ Executed the monthly time clock calculation job for each month until the end of year 2000 (February – December 2000)
- ❑ Repeated steps 2 – 5 for two additional years (2001 – 2002)
- ❑ Repeat steps 2 – 4 for one additional year (2003)
- ❑ Executed a monthly time clock calculation job for February 2003

This approach allowed for days when data loads would not be conducted (e.g., weekends). Time clocks were executed at the conclusion of every daily load in order to incorporate the batch transactions as well as any online updates that may have affected time clock calculations.

During Cycles II and III, the System Test Team repeated the same steps as Cycle I. Each cycle lasted approximately two weeks. ***Figure 4-1: Batch Testing Calendar*** depicts the batch testing execution schedule. It shows the schedule used to process the daily county load files and corresponding daily and monthly time clock calculation modules. The term "Daily" represents a batch processing day, and "Monthly" indicates the execution of the referenced monthly time clock calculation module.

Figure 4--1: Batch Testing Calendar

Monday	Tuesday	Wednesday	Thursday	Friday
Mon Jan 17	**Tue Jan 18**	**Wed Jan 19**	**Thu Jan 20**	**Fri Jan 21**
Cycle 1: Initial Load 2000	**Daily: Jan 10, 2000** **Daily: Jan 13, 2000** **Daily: Jan 18, 2000** **Daily: Jan 21, 2000** **Daily: Jan26, 2000** **Daily: Jan 31, 2000**	**Monthly : Jan 2000** **Daily: Feb 3, 2000** **Monthly: Feb 2000** **Monthlies: Mar - Dec 2000**	**Daily: Jan 10, 2001** **Daily: Jan 13, 2001** **Daily: Jan 18, 2001** **Daily: Jan 21, 2001** **Daily: Jan 26, 2001** **Daily: Jan 31, 2001**	**Monthly: Jan 2001** **Daily: Feb 3, 2001** **Monthly: Feb 2001** **Monthlies: Mar - Dec 2001**
Mon Jan 24	**Tue Jan 25**	**Wed Jan 26**	**Thu Jan 27**	**Fri Jan 28**
Daily: Jan 10, 2002 **Daily: Jan 13, 2002** **Daily: Jan 18, 2002** **Daily: Jan 21, 2002** **Daily: Jan 26, 2002** **Daily: Jan 31, 2002**	**Monthly : Jan 2002** **Daily: Feb 3, 2002** **Monthly: Feb 2002**	**Monthlies: Mar - Dec 2002**	**Daily: Jan 10, 2003** **Daily: Jan 13, 2003** **Daily: Jan 18, 2003** **Daily: Jan 21, 2003** **Daily: Jan 26, 2003** **Daily: Jan 31, 2003**	**Monthly: Jan 2003** **Daily: Feb 3, 2003** **Monthly : Feb 2003**
Mon Jan 31	**Tue Feb 1**	**Wed Feb 2**	**Thu Feb 3**	**Fri Feb 4**
Monthlies: Mar - Dec 2003	**Cycle 2: Initial Load 2000**	**Daily: Jan 11, 2000** **Daily: Jan 14, 2000** **Daily: Jan 19, 2000** **Daily: Jan 24, 2000** **Daily: Jan 27, 2000**	**Monthly : Jan 2000** **Daily: Feb 2, 2000** **Monthly: Feb 2, 2000** **Monthlies: Mar - Dec 2000**	**Daily: Jan 11, 2001** **Daily: Jan 14, 2001** **Daily: Jan 19, 2001** **Daily: Jan 24, 2001** **Daily: Jan 27, 2001**
Mon Feb 7	**Tue Feb 8**	**Wed Feb 9**	**Thu Feb 10**	**Fri Feb 11**
Monthly : Jan 2001 **Daily: Feb 2, 2001** **Monthly: Feb 2, 2001** **Monthlies: Mar - Dec 2001**	**Daily: Jan 11, 2002** **Daily: Jan 14, 2002** **Daily: Jan 19, 2002** **Daily: Jan 24, 2002** **Daily: Jan 27, 2002**	**Monthly : Jan 2002** **Daily: Feb 2, 2002** **Monthly: Feb 2, 2002** **Monthlies: Mar - Dec 2002**	**Daily: Jan 11, 2003** **Daily: Jan 14, 2003** **Daily: Jan 19, 2003** **Daily: Jan 24, 2003** **Daily: Jan 27, 2003**	**Monthly : Jan 2001** **Daily: Feb 2, 2001** **Monthly: Feb 2, 2001** **Monthlies: Mar - Dec 2001**

Monday	Tuesday	Wednesday	Thursday	Friday
Mon Feb 14	**Tue Feb 15**	**Wed Feb 16**	**Thu Feb 17**	**Fri Feb 18**
Cycle 3: Initial Load 2000	**Daily: Jan 12, 2000** **Daily: Jan 17, 2000** **Daily: Jan 20, 2000** **Daily: Jan 25, 2000** **Daily: Jan 28, 2000**	**Monthly : Jan 2000** **Daily: Feb 2, 2000** **Monthly: Feb 2000** **Monthlies: Mar - Dec 2000**	**Daily: Jan 12, 2001** **Daily: Jan 17, 2001** **Daily: Jan 20, 2001** **Daily: Jan 25, 2001** **Daily: Jan 28, 2001**	**Monthly: Jan 2001** **Daily: Feb 2, 2001** **Monthly: Feb 2001**
Mon Feb 21	**Tue Feb 22**	**Wed Feb 23**	**Thu Feb 24**	**Fri Feb 25**
Monthlies: Mar - Dec 2001	**Daily: Jan 12, 2002** **Daily: Jan 17, 2002** **Daily: Jan 20, 2002** **Daily: Jan 25, 2002** **Daily: Jan 28, 2002**	**Monthly : Jan 2002** **Daily: Feb 2, 2002** **Monthly: Feb 2002**	**Monthlies: Mar - Dec 2002**	**Daily: Jan 12, 2003** **Daily: Jan 17, 2003** **Daily: Jan 20, 2003**
Mon Feb 28	**Tue Feb 29**	**Wed Mar 1**	**Thu Mar 2**	**Fri Mar 3**
Daily: Jan 25, 2003 **Daily: Jan 28, 2003** **Monthly : Jan 2003**	**Daily: Feb 2 2003** **Monthly: Feb 2003**			

Functionally, batch testing consisted of examining the loading of a combination of CalWORKs, TANF, Child Support Reimbursement, Welfare-to-Work, Under 10 Grant, Supportive Services, and Non-California program type participation, exceptions and changes to an individual's identifying information (e.g., Date of Birth, Case Serial Number). Tests were performed to validate inserting new records, making modifications to, and deleting existing records. Other testing focused on sending bad data (data that does not match existing WDTIP data types or is null) to the TRAC application. Daily time clocks were then run on combinations of the above data and validated against the expected results.

Additional batch testing focused on validating monthly time clock functionality. Tests were conducted for individuals that had modifications or additions to existing data (for that month) as well as individuals that had no modifications or additions. Data loads were interrupted to validate TRAC application restart logic.

4.4.4.3 Report Files

The System Test Team conducted two cycles of report file testing. This testing consisted of validating output from the four report files generated by the TRAC application and testing the report file program functionality. Report files were validated on three criteria: accurate counts (numbers of individuals qualifying for any/all of the four categories of reports), accurate reporting of individual-specific information and correct time clock calculations.

There were two major steps in testing the report files. First, the data necessary for the report files was unloaded from the TRAC database into a flat file on the mainframe. Subsequently, the report program was executed against the flat file. It performs calculations on the data and creates the output files that will be sent to the counties. The output files were then validated against the expected results of the scripts.

4.4.4.4 Database

The System Test Team conducted two cycles of database testing. Database testing focused on the testing the requirements presented in the *Technical Requirements* section of the **Updated Business Requirements v06** deliverable. Tests included validating multiple users simultaneously attempting to update the same record and attempting to add a child record for an individual that did not have a corresponding parent record (e.g., adding an out-of-state record, non-California, for an individual that did not exist in the TRAC database).

4.5   Processes

This section focuses on providing information on key processes used during system testing including:

❑   Test scenario execution

❑   Incident tracking

❑   Configuration management/software migration

4.5.1 Test Scenario Execution

Each tester was assigned to specific scripts for execution. Testers printed out the system test documentation from the STS detailing the scenario number, scenario name, script number, script name, step numbers, step instructions and expected results.

Execution of test scenarios was performed on a script-by-script basis. Testers began the execution process by reviewing the program specifications associated to the functionality being executed and tested by the scripts. Then the tester followed the detailed steps contained within each script. After each step was completed, actual results were documented on a hard copy of the test script. The tester then transferred those results to the STS and the PTS (if an incident was discovered) using the expected results to determine whether the scenario, script and steps had been successful.

4.5.2 Incident Tracking

Incidents were identified when actual results differed from the expected results of a given script. Any discrepancies that were identified, were cross-checked to ensure that there were no errors in the data or in the execution of the script. If the discrepancy passed this test, then it was documented as an incident in both the STS and the PTS. The PTS was used by the Application Team as a means of communication between the System Test Team and the Application Team. The actual incident was documented in the PTS; whereas, the test result was documented in the STS.

The System Test Team Lead generated daily reports from the STS, identifying scenarios and associated scripts that did not contain a "pass" status. This report was distributed to Project Management and WDTIP Team Leads attending the daily WDTIP Status Meeting.

At the daily System Test Meeting, the System Test and Application Team Leads reviewed the outstanding, completed and newly created incident forms during the meeting. Then the System Test Team Lead:

❑ Reviewed the successful test results, including successful re-tests

❑ Presented the incidents for validation – as a group

❑ Classified the incident as a change or fix

❑ Informed appropriate team members of the incidents

❑ Returned incidents determined to be invalid for discussion

At the conclusion of the Daily Status Meeting, incidents were updated in the PTS by the testers. The appropriate Application Team Lead reviewed the incidents, attached the Program Change Requests (PCR) to them and assigned them to a programmer for resolution (change or a fix). The incident was then worked on by the assigned programmer and then peer tested by another Application Team member. If an incident implied a change in the functionality, then the corresponding specifications were also updated. The fix or change was then migrated to the system test environment and re-submitted to the System Test Team for testing. These procedures are described in more detail in *Section 3.1.1, Change Request Procedures* of the **Completed Source Modules**

- Unit Test deliverable and *Section 5.3, Configuration Control Process* of the **Configuration Management Plan** deliverable.

4.5.3 Configuration Management/ Software Migration

Migration procedures were an integral part of the configuration management process and were used to track and deploy any changes that occurred in the application code as part of resolving system test incidents. These procedures ensured that the system test environment received updated versions of the application in a predictable and a stable manner.

The source of all migration requests was the incidents identified during system test. These incidents resulted in changes to the programs, which were carried out in the development environment. Once the programs were changed and tested, they were put into a reference library (through the Check-in process). This reference library was used as a source for all the programs that were deployed in the system test environment. The promotion of programs into the system test environment was determined jointly between the Application Team Lead, Tech Support Team Lead and the System Test Manager.

In order to facilitate the migration of programs, a migration request was created. This request identified all the incidents and the corresponding PCRs to be promoted to the system test environment. This included the source code, copybooks and the relevant JCL. The request was handed over to the deployment team, who promoted these objects to the system test environment. The deployment team performed impact analysis to determine the modules impacted and recompiled the impacted programs. Once the deployment of these programs was completed, the status of the incidents was updated to "System Test" to indicate that these incidents were ready for retest. The Testing Manager at the daily System Test Team meeting communicated migrations to the System Test Team.

Migrations occurred at weekly intervals. More frequent migrations were implemented when urgently needed (an emergency). The migration procedures are addressed in more detail in *Section 5.4, Software Control Process* of the **Configuration Management Plan** deliverable.

4.6   Findings

This section provides a summary of the system test findings and is divided into four subsections:

- ❑   Test coverage
- ❑   Test completeness
- ❑   Incidents
- ❑   Final results

4.6.1 Test Coverage

All areas of the TRAC application were tested using the 13 scenarios developed to test the technical and business requirements detailed in the **Updated Business Requirements** deliverable. Two cycles of testing were completed for online, reports and database and

three cycles were completed for batch. (Please see *Section 4.4.1.1, Scenarios and Scripts* for additional details). Please refer to the STS to view the scenarios, scripts and steps used for online, batch processing, reporting and database testing.

4.6.2 Test Completeness

The *Scenario Executions Report,* in *Appendix I A* of this document, provides the details on the number of executions performed for each script. Every script was executed at least twice, sometimes more.
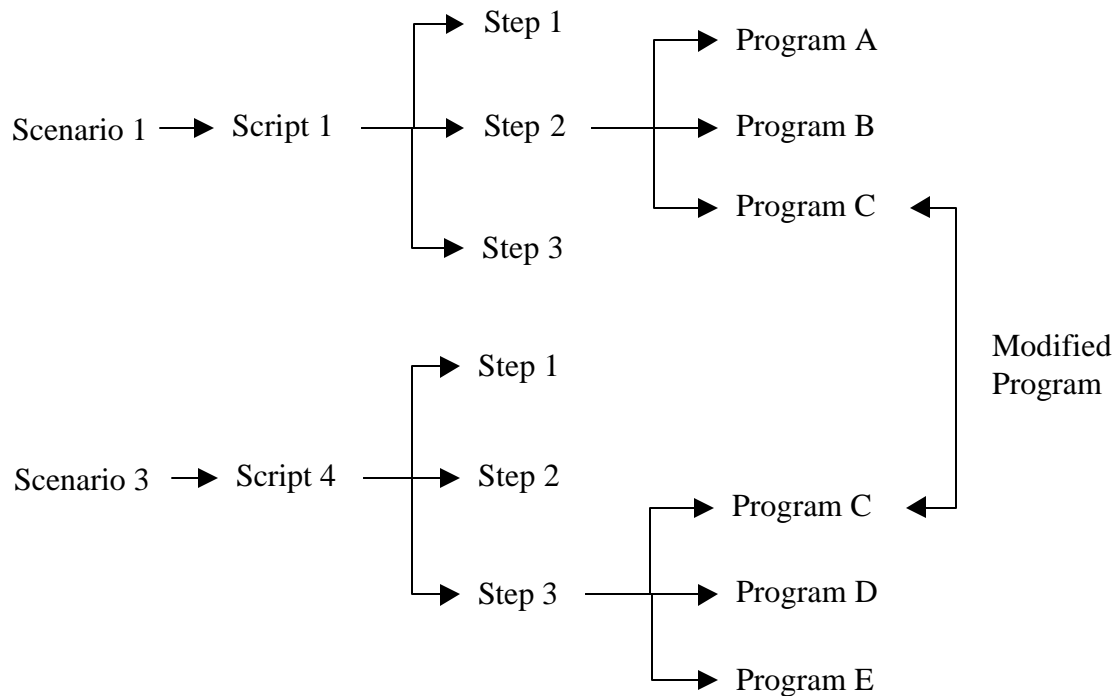
In addition, the System Test Team conducted a thorough regression test on the entire TRAC application (for every program) at the conclusion of system test activities. The regression test was conducted to ensure that changes made to the system did not introduce any new incidents or issues. For a complete listing of scripts that were regression tested, please refer to the report, *Regression Testing Scenarios,* in *Appendix XII* of this document.

This ongoing process of regression testing was used anytime changes were made to the TRAC application. One example of thorough regression testing occurred after the incidents identified in Cycle III were fixed and tested. Smaller-scale regression testing occurred when a program or system modification was migrated to the system test environment. These smaller-scale regression tests validated the correctness of the modified program (or system modification) as well as validating that the changes that were made did not adversely affect other parts of the system.

The figure below illustrates how scripts were chosen for "smaller-scale" regression testing and is based upon the affected program. When an incident was identified during system testing, the Application Team Lead and the Developer (program) identified the programs impacted by the modification to correct the incident. The programs that were modified during incident resolution were queried against the STS to determine the scripts impacted by the program modification. (A table in the STS identified the relationships of scenarios, scripts and steps with the programs it accessed (when it was being executed).

In the example below, Program C was modified to resolve an incident. Scenario 1, Script 1, Step 2 and Scenario 3, Script 4, Step 3 accessed Program C. In order to fully test that changes made to Program C did not affect other previously stable modules, both Scenario 3, Script 4 and Scenario 1, Script 1 were fully regression tested.

Figure 4-2: Overview of Small Scale Regression Testing

```
Scenario 1 → Script 1 → Step 1 → Program A
                        Step 2 → Program B
                        Step 3   Program C ←┐
                                            │
                                     Modified
                                     Program
Scenario 3 → Script 4 → Step 1      Program C ←┘
                        Step 2
                        Step 3 → Program C
                                 Program D
                                 Program E
```

Please refer to *Section 3.4, Regression Testing Procedures* of the **Completed Source Modules/Unit Test** deliverable for more information on the processes used for regression testing.

4.6.3 Incidents

Any discrepancies found during script execution were documented as incidents in the PTS. Incidents were categorized according to the functional area being tested and then prioritized according to the categories identified in the **Configuration Management Plan.** Overall, the majority of the incidents were categorized as defects in the functionality of the TRAC application. Most of the incidents were categorized as high or medium priority. The figure below provides a summary of the incident types and corresponding online or batch cycle number.

Figure 4-3: Incident History
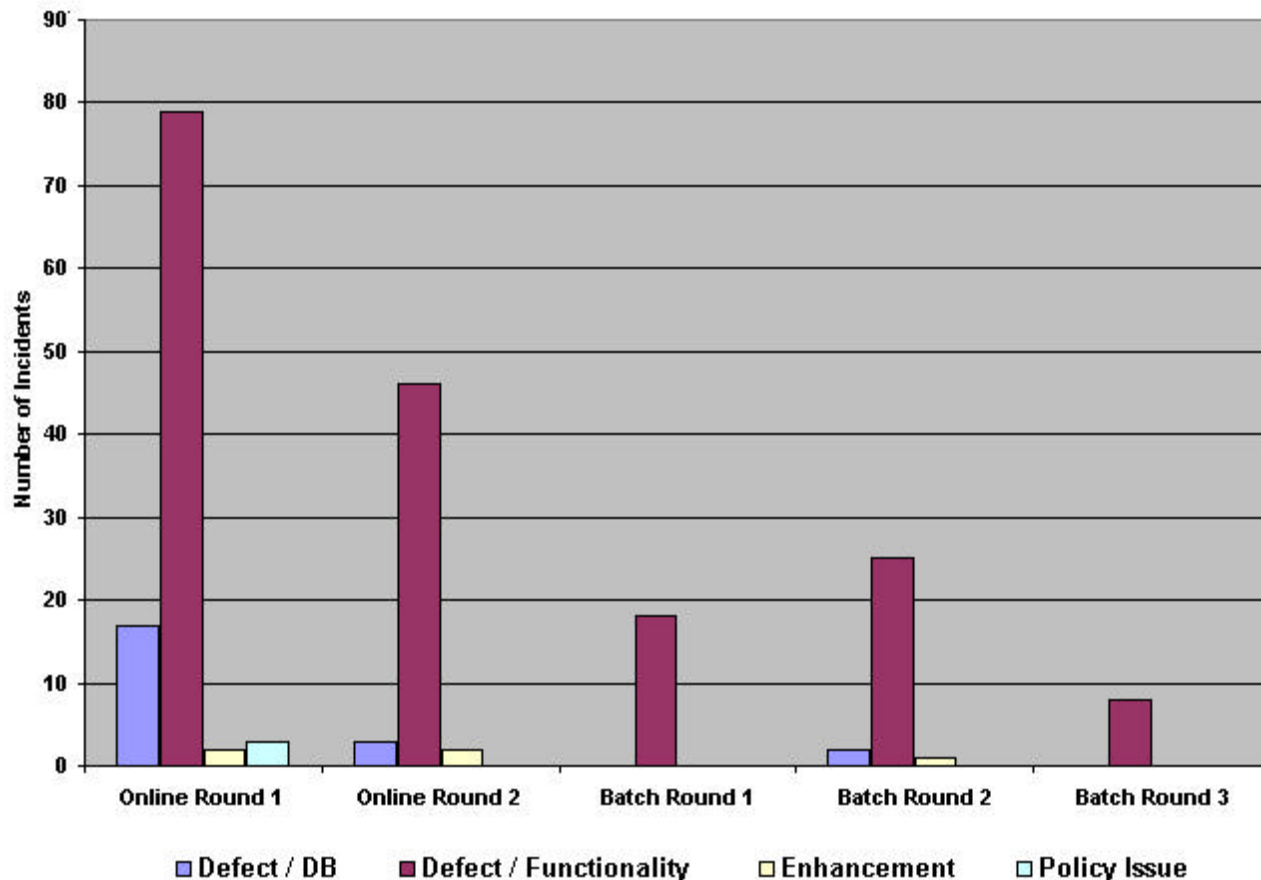
Note: 6 defects in functionality found during round 1 of online and 10 defects in functionality found during round 2 of online were timeclock-related

4.6.3.1 Incident Priority Analysis

For online testing, urgent or high priority incidents typically involved incorrect calculation of time clocks, inability to add, modify and/or delete a record through online updates, and/or incorrect displays of time clock/program participation data on the TRAC application screens. Medium and low priority online incidents consisted mostly of changes to error messages or suggested enhancements.

For batch testing, urgent or high priority incidents consisted of incorrect time clock calculations and data not loading correctly. A smaller percentage of incidents were found to be of medium or low priority status and were related to issues with the batch timestamps and user access issues.

The table below summarizes the priorities and counts of all incidents by focus area.

Table 4-3: Incident Priority and Count by Focus Area

	FOCUS AREA				
Priority	**Online**	**Batch**	**Reports**	**General**	**Totals**
Urgent	0	7	0	0	7
High	37	60	0	0	97
Medium	64	11	0	0	75
Low	47	1	0	0	48
Total	**148**	**79**	**0**	**0**	**227**

4.6.3.2 Incident Type Analysis

Incidents were categorized in one of four ways: defect in the application functionality; defect in the database, enhancement, or policy issue. As stated above, the majority of the incidents were categorized as being defects in the TRAC application functionality. The table below summarizes the types and counts of incidents documented during system testing by focus area.

Table 4-4: Incident Type and Count by Focus Area

	FOCUS AREA				
Incident Type	**Online**	**Batch**	**Reports**	**General**	**Totals**
Defect / Functionality	127	70	0	0	197
Defect / Database	14	8	0	0	22
Enhancement	4	1	0	0	5
Policy Issue	3	0	0	0	3
Totals	**148**	**79**	**0**	**0**	**227**

4.6.3.3 Incident Status

All incidents identified during system testing were closed by the conclusion of system test with the exception of Incident #331. This incident required a change to be made to B3BSC010 to accurately reflect an individual's time clock information. This was not completed during the course of system test since B3BSC010 was being modified to accommodate the SCI interface enhancement, which had been requested by SCI after the start of system test. A decision was made by WDTIP Project Management to combine the resolution of this incident with the enhancement required to the program in order to comply with the updated SCI requirements. This incident was categorized as a defect in the TRAC application functionality and was given high priority. The corresponding change to the design of B3BSC010 was determined to be outside of system test activities since these requirements were addressed by SCI after the start of system test. For more details on Incident #331, please reference the *4.6.4.2 Failed Scripts* subsection.

The *Script Incident Cross Reference Table* provided in *Appendix I B* (of this document) provides a list of each incident identified during system test, along with its associated scenario, script and step number. For detailed information about specific incidents (including incident resolution) see the PTS.

4.6.4 Final Results

The results of system testing are outlined in the *Final Testing Status Report* within *Appendix I C* (of this document) and can be summarized as follows:

- ❑ 162 of the original 178 scripts passed

- ❑ Eight scripts were reclassified as invalid, thereby reducing the total script number to 170

- ❑ Eight of the 170 scripts failed

4.6.4.1 Invalid Scripts

Eight scripts resulted in a status of invalid. Five of these scripts were originally written to test the ENVR screen. This screen was created to assist testers with navigation to the testing environment. Because the screen will not be used in production, the five scripts testing this screen were removed from the system testing scenarios. The last three scripts were given a status of invalid because of duplication. Initially, four separate scripts were written to test the initial load. It was decided that one script would be adequate to fully test this function; therefore, the remaining three scripts were deemed invalid.

4.6.4.2 Failed Scripts

The eight scripts that failed were all linked to the unresolved Incident #331, discussed in *Section 4.6.3.3, Incident Status*. This incident required a change to be made to B3BSC010 to accurately reflect an individual's time clock information. The testing of the scripts that required this correction could not be completed during system test since B3BSC010 was being modified to accommodate the SCI interface enhancement. The fix will be made in conjunction with the changes required for interfacing with SCI. Once the fixes have been made to the program, the incident will be migrated following the promotion model outlined in the **Configuration Management Plan** deliverable. After the incident has been migrated to the system test environment, regression testing will be performed to ensure that the changes made to the program will not adversely affect other WDTIP system functionality.

5. User Acceptance Test Plan

5.1   Definition

For the purposes of WDTIP, UAT is defined as the process by which representative users test the system screens, navigation and functionality.

5.2   Objective

The objective of UAT is to validate the functionality and general use of the WDTIP system as it pertains to the user's business needs. The UAT Team will consist of representatives from the eight systems from which data will be converted, as well as from CDSS. The UAT Team will validate that the WDTIP system:

❑   Meets the business needs of the user and assists the user in performing his/her duties efficiently and effectively

❑   Operates correctly with county-specific data (if data is available)

Specifically, the UAT Plan:

❑   Outlines issues affecting UAT by identifying assumptions and constraints

❑   Provides an overview of the WDTIP UAT strategy and documents the approach to planning, preparing, executing and summarizing the results of UAT. This includes identifying tasks, roles and responsibilities and a milestone schedule

❑   Describes the approach to developing scenarios and scripts and includes the scripts as an appendix

❑   Discusses the recruitment and qualifications of the UAT Team

❑   Details the facility requirements and procurement

❑   Outlines the requirements for development and maintenance of the WDTIP testing region

❑   Outlines the requirements for generating data to support the scenarios/scripts

❑   Provides the acceptance criteria for UAT sign-off

❑   Describes the scenario validation meeting, UAT overview session and WDTIP system training

❑   Describes all the activities and processes in place for testing and retesting, including logging, tracking and fixing/resolving incidents and issues

❑   Presents the approach for providing daily status reports and summarizing final UAT results

5.3   Assumptions & Constraints

This subsection documents significant assumptions that have been used to guide the development of this UAT Plan. Constraints representing potential barriers to the

successful implementation of this Plan have also been identified. Both are included below.

5.3.1   Assumptions

❑ State staff will lead UAT activities, with support from the WDTIP Implementation and Application Teams.

❑ Counties will be able to commit one representative from each of the eight systems from which data will be converted to participate on the UAT Team for the duration of the defined schedule.

❑ CDSS will be able to commit two representatives (one from eligibility and one from employment services) to participate on the UAT Team for the duration of the defined schedule.

❑ Each UAT Team member will be able to spend approximately three weeks in Sacramento (over two separate visits) and a few days' worth of time at their local sites for UAT activities.

❑ UAT Team members will have the authority to represent county or consortia in the signing off of UAT scenarios/scripts.

❑ UAT Team members will have the following skills and knowledge: knowledge of CalWORKs program policies; understanding of the relationship between program participation and TANF, CalWORKs and WTW time-on-aid limits; strong analytical skills and basic computer skills.

❑ The WDTIP Team, with the review and approval of the UAT Team, will be able to identify and develop scenarios and scripts that include a representative sample of the State's welfare business cases as they relate to the WDTIP system. These scenarios/scripts will cover a broad enough set of cases to provide confidence that the system, as built, satisfies the business needs of users.

❑ The UAT Team will be able to test the scripts within the two-week time period designated for testing.

❑ System test will be completed prior to UAT.

❑ UAT will be conducted at a training facility in Sacramento.

❑ The WDTIP Team will generate test data to support the scripts developed, as well as to support the training curriculum used specifically for training the UAT Team. No live county data will be used for UAT.

5.3.2   Constraints

❑ UAT Team members will not be able to devote three months to UAT activities. (Three months represents the time period to conduct all WDTIP UAT activities from planning to summarizing results.) In fact, the amount of time that UAT Team members will be able to devote to UAT activities will be quite limited (15-18 days that are not contiguous).

❑ Counties and CDSS may not be able to commit the same resource for the various activities throughout the UAT period.

❑ The limited amount of time UAT Team members will be able to commit to UAT activities will restrict how involved the UAT Team is in developing scenarios and writing scripts.

❑ The number of scripts executed is dependent upon the amount of time UAT Team members can commit to testing.

❑ No counties will be ready to convert data by April 10, 2000 (when UAT starts); therefore, no county data will be used during UAT.

5.4 Strategy and Approach

Although the most comprehensive UAT would require participation from county/consortia and CDSS representatives for the entire UAT period (planning through execution and documenting results) and would include testing of activities beyond system use (such as Help Desk procedures and training curriculum), the WDTIP Team is working within specific resource constraints. To mitigate the constraints, the WDTIP UAT strategy is based on incorporating representative system users (the UAT Team) to participate in the validation of scripts and testing of the WDTIP system screens, navigation and functionality, while addressing the anticipated limited availability of these individuals. For instance, instead of having the UAT Team actually write the scripts used to test the system, WDTIP Subject Matter Experts (SMEs) will draft the scripts and the UAT Team will validate, revise and supplement (if necessary) the already drafted scripts. This will require less time of the UAT Team without excluding them from the process.

5.4.1 Tasks

The overall strategy consists of three stages: preparing for UAT, conducting UAT, and summarizing UAT results. This subsection outlines the tasks associated with each of these activities. Included below, by task, are the task description and responsible party.

Stage 1 – Prepare for UAT (February 7 – April 7, 2000)

❑ **Draft Test Scenarios and Scripts** – WDTIP SMEs, with support from the WDTIP Implementation Team, completed the first draft of test scenarios/scripts. Approximately 100 scripts, encompassing 14 focus areas (or scenarios), have been drafted to ensure that all variables within the screens are covered, that the time clocks calculate accurately and the information displays correctly. Please see the *5.5 Scenarios and Scripts* subsection for more information.

❑ **Secure Testing Facilities –** The WDTIP Implementation Team secured the California Health and Human Services Agency Data Center (HHSDC) Training Center for testing. The facilities needed to have a minimum of 12 terminals with MEDS and wide area network (WAN) access, and a projection screen. Please see the *5.7 Facilities and Equipment* subsection for more details.

❑ **Identify and Confirm UAT Team** – The WDTIP Project has solicited system user representatives from the eight main systems from which data will be converted, as well as CDSS representatives, to make up the UAT Team. The UAT Team will be tasked with validating and supplementing test scripts, testing the system to ensure that it meets the business requirements from a user's perspective and retesting the system after all incidents have been fixed and issues resolved. To give the UAT

Team as much time with the first draft of test scripts as possible, the Implementation Team will confirm all members' participation by the beginning of March 2000. For more information, please see the *5.6 UAT Team* subsection.

❑ **Review and Validate Draft Scenarios and Scripts** – As soon as the first draft of scenarios and test scripts has been finalized and all the participants have been confirmed, the WDTIP Implementation Team will forward the scenarios and scripts to each member of the UAT Team for their review, comment and revisions. The UAT Team will be asked to review and provide comments and revisions on the scenarios and scripts prior to meeting in Sacramento for the first week of UAT (the week of March 27, 2000).

❑ **Prepare the Testing Region and Generate Data** – During March 2000, the WDTIP Application Team will prepare the testing region, ensure User IDs are ready and generate test data for scripts. Please see the *5.8 UAT Region* and *5.9 Test Data* subsections for more information.

❑ **Conduct Onsite Scenario/Script Meeting with UAT Team** – Approximately two weeks after the draft scenarios and scripts are sent out, the WDTIP Implementation Team will ask that all comments and revisions be returned. The WDTIP Implementation Team will consolidate and respond to comments, make appropriate revisions, develop supplemental scenarios/scripts (if necessary) and prepare to meet with the UAT Team. During the UAT Team's first visit, the UAT Lead and members of the UAT Team and the WDTIP Implementation Team will discuss the comments, revisions and supplemental scenarios/scripts with the intention of obtaining validation from the UAT Team. Please see the *5.11 UAT Overview and Training* subsection for more details.

❑ **Provide UAT Overview and Train the UAT Team** – Also during the first visit, the UAT Lead will provide the Team with a UAT overview (which will include an overview of the UAT process and a time clock refresher) and the WDTIP Implementation Team will train the UAT Team on the use of the system. The system training will be conducted using the **WDTIP Training Curriculum.** Please see the *5.11 UAT Overview and Training* subsection for more details.

❑ **Conduct Remote Testing** – The week between the first visit and testing, the UAT Team will execute specific scripts developed to test connectivity and performance (i.e., that the time it takes to navigate from one screen to the next mirrors what the UAT Team members would find in MEDS). The UAT Team will be instructed during their first week, and WDTIP Team members will be available to support them during actual testing.

Stage 2 – Conduct UAT (April 10 – 21, 2000)

❑ **Test the System** – Starting April 10, 2000, the UAT Team will return to Sacramento for the second time to test the system. Led by the UAT Lead, with assistance from the WDTIP Implementation and Application Teams, the UAT Team will test the system by following the test scripts and logging all issues and incidents found directly into the PTS. The UAT Lead will conduct a daily meeting with the UAT Team and representatives from the WDTIP Implementation and Application Teams to discuss these incidents and issues to ensure that the Application Team knows

exactly what fixes need to be made. See the *5.12 Testing and Retesting* subsection for further details.

❑ **Fix Incidents and Resolve Issues** – During the testing timeframe, the WDTIP Application Team will dedicate evenings to fixing incidents and resolving issues so that the UAT Team can retest those scripts during the testing timeframe. The fixes and retests are expected to be an iterative process throughout the two-week period. For more information, please see the *5.12 Testing and Retesting* subsection.

Stage 3 – Summarize UAT Results (April 24 – 28, 2000)

❑ **Summarize UAT Results** – During the final week of UAT, the WDTIP Implementation Team will complete a summary of UAT findings to be submitted with the **User Acceptance Test Sign-off** deliverable. More information on summarizing the findings can be found in the *5.13 UAT Results* subsection.

5.4.2   Roles and Responsibilities

Many different entities and individuals must be involved in the completion of the identified tasks to ensure the successful and complete testing of the WDTIP system. Responsible agencies and individuals are detailed below.

5.4.2.1 WDTIP Team

Although the WDTIP Implementation Team will be responsible for many of the activities associated with UAT, the WDTIP Application Team will also provide critical support.

The **WDTIP Implementation Team** will support the UAT effort by:

❑ Drafting scenarios and scripts

❑ Securing and preparing testing facilities

❑ Creating and submitting the UAT Plan

❑ Generating mock data to support the scenarios/scripts

❑ Consolidating comments/revisions on the scenarios and scripts by the UAT Team

❑ Training the UAT Team on the WDTIP system

❑ Supporting the UAT Lead and the UAT Team during testing/retesting activities

❑ Managing communications between stakeholders (including developing daily status report)

❑ Summarizing test results

❑ Developing and submitting the **User Acceptance Test Sign-off** deliverable

The **WDTIP Application Team** will support the UAT effort by:

❑ Creating a testing region and associated user IDs

❑ Preparing the testing facilities

❑ Supporting the UAT Lead and the UAT Team during testing/retesting activities

❑ Running batch programs as needed

❑ Fixing incidents and resolving issues

5.4.2.2 Counties/Consortia

The WDTIP Team will work closely with county and consortia representatives that make up the UAT Team in all UAT activities.

The **UAT Team** will support the UAT effort by:

❑ Providing comments on scenarios and scripts

❑ Participating in the scenario validation meeting and signing off on final scenarios/scripts

❑ Participating in the UAT overview and WDTIP system training

❑ Testing and retesting the WDTIP system (including testing at their local sites, "remote testing")

❑ Recording incidents and issues

❑ Participating in the daily UAT meeting to discuss issues and incidents

❑ Summarizing UAT results

❑ Participating in the debriefing session

5.4.2.3 State Agencies

Although the WDTIP Implementation and Application Teams will take care of most of the planning and preparation, State staff on the WDTIP Team will lead UAT efforts.

The **UAT Lead** will support the UAT effort by:

❑ Confirming UAT participants

❑ Developing UAT acceptance criteria

❑ Leading the scenario validation meeting

❑ Providing the UAT overview

❑ Leading the testing/retesting activities, including the daily UAT meeting

❑ Debriefing the UAT Team

5.4.3 Schedule

The following table provides timeframes for the high level tasks provided in the *5.4.1 Tasks* section above.

Table 5-2: Milestone Schedule

Task	Responsible Party	Timeframe
Stage 1 – Prepare for UAT		
Draft Test Scenarios and Scripts	SMEs Implementation Team	February 2000 – March 3, 2000
Secure Testing Facilities	Implementation Team	February 2000 – March 3, 2000
Identify and Confirm UAT Team	UAT Lead Counties/Consortia	February 2000 – March 3, 2000
Provide UAT Team with Draft Scenarios and Scripts	Implementation Team	March 3, 2000
Prepare Testing Region and Generate Data	Application Team Implementation Team	February 2000 – April 7, 2000
Review Scenarios and Scripts	UAT Team	March 7, 2000 – March 21, 2000
Prepare Testing Facilities	Implementation Team Application Team	March 27, 2000 – April 7, 2000
Validate Scenarios and Scripts	UAT Lead UAT Team	March 28, 2000
Provide Overview and Train the UAT Team	UAT Lead Implementation Team	March 29, 2000
Conduct Remote Testing	UAT Team Implementation Team	April 3, 2000 – April 7, 2000

Task	Responsible Party	Timeframe
Stage 2 – Conduct UAT		
Test the System/Log Incidents and Issues	UAT Team HHSDC Application Team Implementation Team	April 10-20, 2000
Fix Incidents and Resolve Issues	Application Team WDTIP Management	April 10-20, 2000
Debrief	UAT Lead	April 21, 2000

Task	Responsible Party	Timeframe
Stage 3 – Summarize UAT Results		
Summarize UAT Results/Develop **User Acceptance Test Sign-off** Deliverable	UAT Team Implementation Team	April 17-28, 2000

5.5   Scenarios and Scripts

Scenarios and scripts are tools that will be used by the UAT Team. Scenarios focus on specific functional areas of the system and are comprised of individual scripts. Scripts provide detailed information about the situation to be tested as well as step-by-step instructions to guide a tester through the script. Scripts also provide the expected results of each step taken so that the tester can verify that the system functions and displays information correctly. Scripts have been designed to test variables that may be encountered when using the system.

Scenarios and their scripts have been developed to cover the most common situations that users encounter while performing their duties in a real life situation. Scenarios used for UAT have been developed to test the system from the user's perspective and should not be considered a system test, which is designed for a different function. For more information on system test scenarios, please see the *System Test Plan*, *Appendix 2* of the **Completed Source Modules/Unit Test** deliverable. Upon completion of all UAT scenarios, the tester should be confident that the system functions as it is designed. By carefully following a step-by-step outline and verifying specific expected results, the UAT Team will be able to certify that the system requirements have been successfully met.

Each scenario addresses a broad area of the system. These areas were identified during a review of the **Updated Business Requirements** deliverable. Once the scenarios were identified, individual scripts were designed to ensure a complete test of each scenario. The number of scripts required to test a scenario varied depending on its complexity. Scripts were written to encompass either a single situation or incorporate multiple situations to simulate actual welfare department case information.

The table below documents the identified scenarios and the number of scripts contained in each scenario.

Table 5-3: Scenario and Script List

Scenario Number	Scenario Name	Number of Scripts
1	System Navigation	4
2	Security	3
3	Individual Inquiry	2
4	Program Participation	2
5	Diversion	4
6	Child Support Reimbursement	2
7	Supportive Services	3
8	WTW Plan	9
9	Time Clocks	50

Scenario Number	Scenario Name	Number of Scripts
10	Non-California Participants	6
11	Reports	2
12	File Extract Loads	4
13	Conversion	2
14	Remote Testing	4
	Total Scripts	**97**

5.5.1 Strategy

In developing scenarios and scripts, the following considerations were made that affected the approach:

❑ Each scenario must meet the WDTIP system business requirements from a user's perspective. SMEs reviewed the **Updated Business Requirements** deliverable to determine the best way to ensure each individual situation would be tested. This identified the focus for each script. Please see the *Business Requirements and Scenario-Script Association, Appendix II* (of this document) for details.

❑ Normally, system users would write the UAT scripts. Because this was not feasible due to limited availability of system users, scripts were written by contracted SMEs with extensive experience at all levels of eligibility work and no prior WDTIP experience. This was done to ensure that the scripts were written to simulate actual welfare situations that would address the needs of a typical user.

❑ The availability of UAT Team members was identified as a possible constraint. Therefore, scripts were designed to ensure an adequate test of the system within the limited amount of time the Team members would be available. It is assumed that the team will consist of ten members for a total of three weeks. A prioritization structure was incorporated into the script design to allow for a possible schedule change. Scripts were assigned a priority of high, medium, or low. An adequate test could be accomplished by completing only the high priority script. Please see the *5.5.4 Prioritization* subsection for more detail on priorities.

❑ The system's security protocol is designed to allow designated users update access to only those updateable screens that require information that cannot be sent to WDTIP in the batch file process. To adequately test the system security, a variety of different county welfare department security structures had to be considered. Different profiles representing update or inquiry access will need to be created for these representations. Scripts were written to test these security functions.

5.5.2 System Functions

The system performs certain functions, without user involvement, that affect areas such as security, daily batch runs, time clock advancements, county extract files, and reports. Although the user does not initiate these functions, they are a part of the system that will need to be tested because they affect what the user sees. Some of these system functions

will need to be simulated to provide an adequate test. This will allow the tester to view the system performance over a condensed period of time. These areas are outlined below.

- ❑ **Security –** The WDTIP system consists of 20 screens; 16 are inquiry only and four screens allow online updates. Inquiry screens will be the most commonly accessed and will display time clock related information and calculations. All users who have inquiry access to MEDS will have inquiry or "read only" access to all the screens in the WDTIP system. The four update screens, however, can only be accessed by designated users within designated counties. Once a county has converted to the WDTIP system, their designated update users will only be able to enter information into the update screens that their county is unable to provide through the standard file extract process. To test the varying levels of access, each tester will be provided with three different log-on IDs. The log-ons will simulate users with inquiry only access, users with update access to some of the update screens, and users with update access to all of the screens.

- ❑ **Daily Batch Runs –** In production, the system executes a daily batch process each night and recalculates the time clocks based on any newly acquired information. Simulation of the daily batch process must be planned to allow for processing data entered into the system via the update screens by the testers. The daily batch process must take place at pre-determined intervals during the test process. This will allow the tester to move through the scenarios without having to wait until the next day to validate time clock recalculations, as they would in a live environment.

- ❑ **Time Clock Advancements –** To ensure an adequate test of the system's time clock calculations, the ability to access the system looking at both the present date and dates in the future is required. This will be accomplished by advancing or "rolling" the clock forward to pre-determined points in time and viewing the new time clock calculation results of previously entered information. It was necessary to identify these points in time, before writing the scripts, to enable expected results to be calculated and included as part of each script.

The following table defines the clock advancements and why they are required.

Table 5-4: Time Clock Advancement Dates

Simulated Date	Description
April 2000	This will be the month that the actual UAT testing will take place.
May 2000	This will allow for the simulation of the monthly recalculation of the time clocks to be reviewed.
August 2001	This date was picked to represent a date that would allow for testing the WTW 18/24-month clock. That date would be a reasonable representation for individuals approaching or exceeding this clock. It would also allow for a mid-way test of the 60-month clocks.
November 2003	This will be the last clock advancement. It allows for validation of reports by comparing the report from the previous clock. This date will allow for a final validation of all the time clocks.

❑ **County Extract Files** – The UAT region will be populated with individual client data for the users to execute inquiry and/or update scripts. The system will receive data during testing three different ways. A county extract file will be simulated to electronically populate the UAT test area with initial data. It will also be the method that data is entered at each clock advancement. Another way the system will receive data will be by direct input from each tester while testing scripts involving update screens. The system will also receive information from SCI. The SCI information will consist of the demographic information attached to each participant and will be for display only.

❑ **Reports –** The system will generate four electronic report files to send to each county. Two of these report files will be tested during UAT and two of these report files will not, based on time constraints during UAT and on the complexity of validation. Two of these reports include aggregate numbers (the number of recipients with multiple county involvement and the number of recipients within six months of approaching any of their clocks). Validating the accuracy of these two reports would require that testers manually count all recipients that meet these criteria by county. Instead, the Implementation Team will test these reports after UAT. The reports that will be tested will need to be translated into a format that the UAT Team will be able to read. These translated reports will be provided to the Team following the last clock advancement.

5.5.3 Test

The testers will execute each step of every script and record incidents and/or issues discovered while executing the step or record the successful completion of the script. All testers will test the same scripts simultaneously. This will aid with the identification of potential system problems by providing comparative results. Although each tester will test each script, each tester will be provided with their own set of participants with common demographic information and identical program participation information. Please see the *5.9 Test Data* and *5.12 Testing and Retesting* subsections for details.

5.5.4   Prioritization

The scripts were written to accommodate a UAT Team of ten members, which will be available for a total of three weeks. Due to a potential variance in the number and duration of tester availability, flexibility was incorporated into the script writing process by assigning each script a priority of high, medium or low. In the event that testers are not available for the duration of the defined test period or testers are not able to finish executing all scripts, the Implementation Team will work with the State to select specific scripts based on the following priority assignments.

Table 5-5: Priority Descriptions

Term	Description
High	High priority scripts were defined as scripts that must be tested to ensure an adequate test of the system. All high priority scripts will be specific to a particular batch run or clock advancement. High priority scripts will cover the functionality of medium and low priority scripts, but will not focus on the lower priority areas such as navigation or security.
Medium	Medium priority was defined to create a set of scripts that would be best to test, but the functionality would be indirectly covered in a high priority script and therefore could be pulled from the set if absolutely necessary. As an example, medium priority scripts might focus on a specific exception, compared to exceptions in general that will be adequately covered within the high priority scripts. The medium priority scripts may be specific to a time clock advancement or daily batch run and therefore need to be executed at a specific time in the test schedule.
Low	Low priority would be criteria similar to medium but would focus on items such as navigation or individual inquiry. Those areas will be included in high priority scripts but are not the focus or primary reason for a high priority script to fail. For example, one navigation method might not work correctly but the tester could still access the required screen. Low priority scripts are not tied to a time clock advancement or batch run and could be executed at any time during the test schedule.

5.5.5   Scenario and Script Approval

The finalization and acceptance of the scripts involves a detailed review by four separate groups: SMEs, WDTIP technical experts, State staff, and the UAT Team. This will ensure that the scripts will provide an adequate test and accurately reflect the system design.

SMEs completed the first draft of each script, then forwarded them to additional SMEs who further reviewed them to ensure they accurately reflected the program details represented in each one. This included validation of expected results by checking

regulatory requirements and time clock calculations. WDTIP technical experts then reviewed the scripts to confirm the accuracy of projected navigational and display results as well as to ensure they appropriately reflected approved system design requirements. They also reviewed the scripts to verify that the data and logistical needs required to set up the test environment could be met.

Finally, the scripts will be sent to members of the UAT Team for review and validation. After review, the Team members will have the opportunity to respond with comments. The test schedule requires the team to assemble prior to the beginning of actual testing to discuss, validate and agree upon the appropriateness of all scripts. Corrections and inconsistencies will be identified during this process. If issues arise in this forum that cannot be resolved, they will be raised to CDSS for finaldecision. Appropriate comments and corrections will be incorporated into the scripts as they are identified. There will be one week between the initial UAT Team meeting and the beginning of testing. This week will allow for additional corrections, clarifications, or additions previously identified to be integrated into the scripts prior to the beginning of UAT.

In the event that script errors are identified during the testing process, the script will be corrected as soon as possible. If the script cannot be corrected in time to proceed within a predetermined clock advancement or batch process, every effort will be made to redesign the script to test the same focus area later in the testing schedule. The UAT scripts have been attached as *Appendix III, UAT Scripts.*

5.6   UAT Team

To meet the objectives of UAT, which is to validate that the WDTIP system meets the business needs of the user, representative users must test the system. Therefore, the WDTIP Team has requested that one county and/or consortia representative from each of the eight systems from which data will be converted and two CDSS representatives participate in UAT activities. These ten people will comprise the UAT Team and include representatives from the following:

- ❑ Interim Statewide Automated Welfare System (ISAWS)
- ❑ Los Angeles Eligibility, Automated Determination, Evaluation and Reporting (LEADER)
- ❑ Welfare Case Data System (WCDS)
- ❑ Merced County
- ❑ Riverside County
- ❑ San Bernardino County
- ❑ Stanislaus County
- ❑ Ventura County
- ❑ CDSS

5.6.1   Communications

On February 17, 2000, correspondence from the SAWS Project Deputy Director advised County Welfare Directors of the Project's plan to begin UAT activities and solicited one representative from each of the eight county/consortia systems mentioned above and two representatives from CDSS. County Welfare Directors were informed of the UAT Team objectives, qualifications and proposed schedule. The *UAT Request for Participation Letter*, *Appendix IV* (of this document), was also carbon copied to CDSS, Consortia Liaisons and Managers, County Contacts and the California Welfare Directors Association (CWDA) Information Technology Liaison.

The WDTIP Implementation Team has followed-up this correspondence by calling some of the WDTIP County Contacts to field questions about the process and to reinforce awareness of the UAT Team objectives, schedule and qualifications. For a list of the confirmed participants, please reference *Appendix V, UAT Team Roster*. The schedule of activities and the desired qualifications of the UAT Team members are outlined in the subsections below.

5.6.2   UAT Team Activities and Schedule

The UAT Team will be asked to participate in various UAT activities from reviewing test scripts to testing the system and signing off on UAT. Most of the activities will occur in Sacramento from the beginning of March to the end of April 2000. The UAT schedule below represents the UAT Team activities by timeframe and location.

Table 5-6: Proposed UAT Schedule

Activity	Timeframe	Location	Total Time
County/consortia identify and confirm resources for UAT participation	By 3/1/00	Local Office	N/A
UAT Team reviews scenarios and provides comments to WDTIP Project	3/6 – 3/21/00	Local Office	2-3 days
UAT Team validates and finalizes scenarios	3/28 – 3/29/00	Sacramento Project site	2 days
UAT Team receives UAT overview and system training	3/30/00	HHSDC Training Center	1 day
UAT Team conducts remote testing	4/5/00	Local Office	4 hours
UAT Team tests system	4/10 – 4/21/00	HHSDC Training Center	10 days

5.6.3 Desired Qualifications

To ensure the successful testing of the WDTIP system from the user's perspective, it is necessary that each member of the UAT Team have specific welfare program knowledge, a specific level of authority and availability throughout the duration of UAT Team activities. Specifically, the Project has requested that each UAT Team member have the following qualifications:

❑ Understanding of CalWORKs program policies

❑ Understanding of the relationship between program participation and TANF, CalWORKs and WTW time-on-aid limits

❑ Analytical skills

❑ Availability for the entire testing process

❑ Authority to represent county or consortia in the signing off of UAT scenarios

5.7   Facilities and Equipment

The WDTIP Implementation Team is responsible for procuring a facility to conduct UAT and ensuring that the facility has the necessary hardware, software and application access. The WDTIP UAT facility was chosen based upon UAT hardware and software requirements, classroom availability, capacity, location and other equipment.

5.7.1   Location, Availability, Hours and Costs

The HHSDC Training Center meets the necessary hardware, software, room capacity and availability requirements for user testing and has been reserved as the WDTIP UAT site. The facility is located centrally and is available March 29-31, 2000 and April 7-21, 2000. The customary hours are 7:30 a.m. – 4:30 p.m. but extended hours are possible through prior arrangement. HHSDC training staff is on-site at 7:00 a.m. The room charge is $250.00 per day and will be billed against the WDTIP project cost account code after the room rental is completed. If pre-production technical support is required, there is an additional $50.00 charge.

The WDTIP site will be used in conjunction with the HHSDC Training Center to conduct UAT activities, specifically, validating and finalizing scenarios. Team members will validate and finalize approximately 100 UAT scripts prior to UAT testing at the Project site. This UAT activity is expected to take a day and a half to complete and is scheduled to begin the afternoon of March 28, 2000 and be completed March 29, 2000. The Project's Conference Room 114, has been reserved because it accommodates seating for 12-15 and connectivity to the WDTIP system is not required.

5.7.2   Hardware

The HHSDC Training Center accommodates 12 tester workstations and one instructor workstation. Since the WDTIP UAT Plan is recommending a total of ten UAT Team members, this allows extra workstations in the event of equipment failure. Each workstation is equipped with an IBM Pentium-One PC that is configured for printing to an HP Laser Jet 4 located in the classroom. This is important because testers should be able to print WDTIP screens to demonstrate issues/incidents during testing. The UAT Lead will also need a PC to be able to access the PTS to log and manage all incidents, issues and comments reported by the testers.

5.7.3   Software

The HHSDC Training Center is configured to the HHSDC Mainframe, which provides access to the MEDS application. The HHSDC Training Center is also configured to the MEDS Acceptance region (CICSO) which allows access to the WDTIP Acceptance Test region (CICSA) from each individual workstation in the room. CICSA is the WDTIP UAT database that will contain the UAT test data and is the region the UAT Team will use for testing.

The instructor workstation PC also has Microsoft Office 1997 software installed. The UAT Team Lead will need Microsoft Office software to be able to use the PTS for logging and managing issues and incidents.

5.7.4 Site Preparation

The HHSDC Training Center site meets the necessary hardware and connectivity requirements, therefore, site preparation activities will focus on ensuring the UAT Team Lead can access the PTS and that the Project and State staff have appropriate audio-visual (AV) hook-ups. The HHSDC Training Center and the WDTIP Project are on the same WAN. The WDTIP Application Team will ensure the instructor PC is configured with Microsoft Access. Using the WAN for access to the Microsoft Access application enables Project Team members at the WDTIP Project site to have immediate access to incident and issue entries made by the UAT Team Lead in the PTS. If AV equipment is required, it will be provided by the Project and is limited to a Proxima and a laptop.

The table below summarizes the facility reservations for testing purposes.

Table 5-7: WDTIP UAT Facilities Schedule

Facility	Activity	Date Reserved
WDTIP Project Site	UAT Team Members Validate and Finalize Scripts	March 28, 2000 March 29, 2000
	Implementation Team Initial MEDS Connectivity Check	*March 2000 TBD
	Training Center Set-up and Connectivity Check	March 28, 2000
HHSDC Training Center	UAT Overview and WDTIP System Training	March 30, 2000
	Training Center Set-up and Connectivity Check	April 7, 2000
HHSDC Training Center	UAT Week II	April 10, 2000 April 11, 2000 April 12, 2000 April 13, 2000 April 14, 2000
	UAT Week III	April 17, 2000 April 18, 2000 April 19, 2000 April 20, 2000 April 21, 2000

* One additional half day will be scheduled to do an initial MEDS connectivity check. This date is available at no charge and is proposed for the last week of March 2000 to allow time for technical resolution in the event there is an access problem.

5.8   UAT Region

The WDTIP Implementation and Application Teams are responsible for establishing an operational UAT region by mid-March 2000. This includes ensuring the database is populated with the data required to test all UAT scenarios, migrating the TRAC software to the UAT region, and ensuring that the DB2 database and CICS online region are accessible to the UAT Team members at the HHSDC Training Center. The topics below describe the development and design of the UAT database requirements, online requirements, application software, and system administration and connectivity.

5.8.1   Database Requirements

The UAT Team must have a database in the WDTIP system specifically designated for UAT to ensure the integrity of the test data generated and the associated scripts. The WDTIP Application Team has developed a separate DB2 database called the Acceptance Test database (DSISA02) within the DB2 B3TD subsystem. This database provides a controlled environment that simulates the WDTIP production system without compromising the integrity of the test data and associated scenarios.

Data integrity is addressed by the development of a separate DB2 database for UAT that eliminates the possibility of data corruption occurring as a result of sharing a database where development or other test activities are performed. The Application Team has provided limited access to this database to UAT Team members and authorized Project staff.

A controlled environment will be maintained by conducting only UAT activities in the Acceptance Test region. These activities include code migration, program execution, data creation and maintenance, system clock adjustment, and scenario testing. Non-UAT development activities are conducted within the other WDTIP environments (Development, System Test, Performance Test, and Training).

A full technical description of the UAT environment is available in the **System Architecture Model**.

5.8.2   Online Requirements

Each WDTIP Development and Testing environment has a unique CICS region defined to provide an online application capability. The Acceptance Test region (CISISA) has a connection to the MEDS Acceptance Test (CICSO) region and the SCI Acceptance Test (CICINS) region. This connection provides online access to the MEDS and SCI Acceptance Test regions. Entry to the WDTIP Acceptance Test region is gained through the MEDS Acceptance Test region. Connection to the SCI Acceptance Test region provides each individual test CIN with an associated combination of name, birth date and social security number (SSN). This vital statistic information allows the testers to associate the test CIN with the individual information in the WDTIP Acceptance Test database.

The figure below illustrates the *high-level* relationship between the Acceptance Test regions of the three applications: MEDS, WDTIP and SCI.

Figure 5-1: Acceptance Test Online Region Relationships

MEDS	WDTIP (TRAC)	SCI
CICSO Acceptance Test	CISISA Acceptance Test	CICINS Acceptance Test
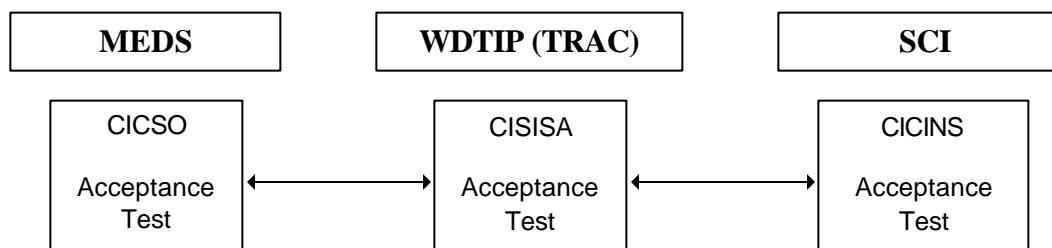
5.8.3   Application Software

The version of the WDTIP application migrated for use in the Acceptance Test region will be the version coming from system test. Application enhancements approved through the Configuration Control Board process follow a migration through the different WDTIP environments (please see the **Configuration Management Plan** deliverable for more information on the process). Any approved software changes will be developed in the Development environment and then migrated to the System Test environment for system regression testing. Once the software is determined to be stable in System Test, it is migrated to Acceptance Test. This means that the version in the Acceptance Test environment might not be as current as the versions in Development or System Test (i.e., when an incident has just been fixed and has not yet been migrated through the system test environment). Version stability is important for UAT because the focus of UAT testers should be on testing the system design and functionality from a user's perspective, not on incidents resulting from system test failures.

5.8.4   System Administration

The UAT Team has access to the WDTIP Acceptance Test environment from the HHSDC Training Center. The Implementation Team will request MEDS User IDs for the UAT Team from the CDHS MEDS Security Coordinator. These User IDs will simulate the production access profiles that will be used in the WDTIP system. Specific profiles will be defined to support the application's inquiry and update functionality.

The User IDs will be used by UAT Team members not only to provide access to the WDTIP Acceptance Test environment, but also to simulate the different security profiles of the WDTIP system users. Three User IDs per UAT Team member will be required to simulate these varying security profiles. The table below contains *sample* IDs intended to represent the three User IDs that will be assigned to each UAT Team member. The table also explains how the profiles relate to the UAT scripts, the level of access and purpose of each profile.

Table 5-8: User ID/Profile Setup for Each Team Member

User ID	Profile Description	Applicability
WDTIP025	Full update capability on all WDTIP update screens.	Specific to the majority of the UAT scripts.
WDTIP026	Partial update capability to specific update screens based upon the data capability of the source system.	Specific to the UAT Security scripts.
WDTIP027	Inquiry access only to all WDTIP screens.	Specific to the UAT inquiry only scripts.

5.9   Test Data

This section details the criteria and process that will be used in the creation and maintenance of test data for the purposes of UAT. This test data will populate the UAT region (please see *5.8 UAT Region* subsection for more information on the testing region). Ideally, UAT would be conducted using converted county data to simulate an environment as close to that of the WDTIP Production as possible. Since actual county data will not be available in time for UAT, test data will be created to simulate the data in the WDTIP Production environment.

Before creating the data to populate the UAT environment, the WDTIP Implementation Team along with SMEs identified the types of data necessary to complete the testing effort as well as the methods by which the WDTIP system will receive the data. The WDTIP system will receive data from five sources in the WDTIP Production environment. Two of these sources, SIS and MEDS, will be used primarily in the initial population of the WDTIP database. The Implementation Team will create test data to simulate four of the five sources. The sources to be simulated during the course of UAT are listed below.

Table 5-9: Data Sources

Data Source	Use of Data	UAT Importance
SIS	SIS will provide baseline data for use in the WDTIP Production environment.	SIS baseline data will be simulated with script data.
SCI	SCI will provide for the linking of CINs to WDTIP records.	SCI provided test CINs to be attached to script individuals.
County/Consortia Systems	Counties/consortia will provide data during initial conversion and updates will occur through daily batch loads.	County/consortia data will be simulated in three data loads subsequent to the initial data load during UAT.

Data Source	Use of Data	UAT Importance
User Generated	Certain data elements will be entered online into the WDTIP system by those counties with update capability.	User generated data is a primary component of UAT. UAT Team members will be updating/generating data within the four update screens throughout UAT.

The data for UAT will be created and organized into four separate data loads. The initial data load will occur just prior to UAT and will populate the test environment with data required for the specified scripts. Three subsequent data loads will be scheduled to occur during the course of UAT. The subsequent data loads will simulate the file extract loads received from the counties' automated systems. There are specific scripts that will be tested after each data load.

5.9.1   Data Requirements

The following outlines the data required to support the functions of UAT. The data requirements have been divided into four sections based on origin.

5.9.1.1 Script Data

Based on the business requirements of the WDTIP system, 42 mandatory county data elements are necessary to accurately calculate the Federal and State 60-month and the WTW 18/24-month time clocks. Each of the data elements falls into one of 13 focus areas. Approximately 100 scripts within the 13 focus areas have been written to ensure the testing of all variables within the WDTIP system. Data will be created to support these scripts, which were developed to represent actual welfare program situations. (Please refer to *5.5 Scenario and Scripts* subsection for more information on the scripts).

Each script involves one individual for which data will need to be created. Welfare program data varies from script to script and will be specific to the individual. A total of 2250 individuals will need to be created: 900 of these individuals will be used to test program specific scripts and the remaining 1350 individuals will be used to test report scripts. It is important to note that the data supporting the report scripts must remain separate from the data created to support the program scripts to ensure its integrity.

Script data, separate from the data detailed above, will be developed to support the execution of the UAT Training Curriculum. Specific information regarding number of scripts necessary and supportive data required will be determined at a later date.

5.9.1.2 Client Identification Number Assignment

SIS and MEDS are built around an identification repository called SCI. SCI supplies and tracks identification numbers or CINs for all individuals that are known to welfare since 1994. It will be necessary to build CINs into our script program data to simulate this process in the WDTIP Production environment. In creating the data, each individual's welfare program data will be attached to a CIN by the WDTIP Implementation Team in an Excel spreadsheet. Once completed, the Excel spreadsheet is then uploaded into the WDTIP system.

Test CINs have been acquired from SCI. The CINs include name, date of birth and SSN. Some scripts contain specific data relating to the age of the individual so it will be necessary to identify CINs that match the appropriate birth dates of the individuals in those scripts. Script data will be attached to a specific CIN creating a viable individual for use in UAT. To support both the program and report scripts, the WDTIP Implementation Team will use approximately 2250 CINs in our data creation. There will be 150 CINs needed per data set and the Implementation Team will be creating 15 data sets (one for each of the ten expected UAT Team members and five backup sets).

Each of the UAT Team members will have their own set of 150 individuals with corresponding CINs to complete UAT. The script data will be the same in each of the sets with the exception of the CIN. By giving each tester their own set of individuals with unique CINs, possible confusion that may occur with multiple users accessing the same individual at the same time should be eliminated.

5.9.1.3 Data Loads

To best simulate the Production environment, the WDTIP Implementation Team has scheduled an initial data load and three subsequent data loads during UAT (to simulate nightly batch). The three data loads were built-in to simulate file extract loads or conversion loads which, after implementation, will be received from counties electronically. It will be necessary to create a data file with script specific, welfare program information for each data load. Test data was created with these data loads in mind. When the test data is created, it will be organized in a manner that will coincide with the data load schedule that parallels the clock advancements noted in the *5.5 Scenarios and Scripts* subsection of this document.

Information required for updates and error processing is documented in each script and will be entered during UAT by the testers themselves. The user-generated data will be incorporated into the database through a simulated batch process. Batch filters the user-generated information into the WDTIP database and recalculates the time clocks for the updated individual.

As a precautionary measure, the initial data load will be copied and this copy will serve as a backup should the need arise. After the commencement of UAT, periodic snapshots of data created and the updates submitted by the testers will be recorded. These snapshots will occur immediately prior to the subsequent data load and will be utilized as a back-up file should problems arise with batch runs, data loads or time clock advancements.

5.9.1.4 User Generated Data

Information required for updates and error processing is documented in each script and will be entered during UAT by the testers themselves. The user-generated data will be incorporated into the database through a simulated batch process. Batch filters the user-generated information into the WDTIP database and recalculates the time clocks for the updated individual.

5.10 Acceptance Criteria

Acceptance criteria are criteria developed by the Project sponsor that outlines acceptable UAT results. CDSS, in conjunction with HHSDC, has developed these criteria based on the business requirements identified during the Joint Requirements Planning (JRP) Session in October 1999.

The UAT acceptance criteria is included as *Appendix XIII, UAT Acceptance Criteria*.

5.11 UAT Overview and Training

The WDTIP Implementation Team is responsible for working with the UAT Team to validate script accuracy and completeness and for providing the UAT Team with an overview and WDTIP system training. The Implementation Team is also responsible for revising and potentially adding scripts based upon input from the UAT Team following validation.

5.11.1 Scenario and Script Validation

The UAT Team will be requested to validate the completeness and accuracy of approximately 100 scripts developed to test the WDTIP system. The scripts must be specific to the system's business and functional requirements and accurately depict testing results for the following areas: System Navigation, Security, Individual Inquiry, Program Participation, Diversion, Child Support Reimbursement, Supportive Services WTW Plan, Time Clocks, Non-California Participants, Report Files, File Extract Loads and Conversion. The WDTIP Implementation Team will forward the scripts to all UAT Team members at the beginning of March 2000. During the first week of UAT activities, the UAT Team will provide the Implementation Team with input regarding accuracy or completeness. It is important to note modifications or additions to the scripts will be limited to those within the current WDTIP system scope and design.

Once the scripts have been validated, the WDTIP Implementation Team will provide the UAT Team with a UAT Overview and with WDTIP system training.

5.11.2 UAT Overview

Because members of the UAT Team may not be experienced in the UAT process or tester roles and responsibilities, the WDTIP Implementation Team will provide the UAT Team with a UAT overview. This overview will provide the UAT Team with the objectives and schedule of UAT activities and describe UAT Team roles and responsibilities. This overview will also provide a general description of the WDTIP system design and functionality (including how time clocks are calculated) to ensure testers understand the scope of the system's business and functional requirements.

The UAT Team will also be working with two forms developed for UAT testing, the *Incident/Issue Identification Form, Appendix VI* and the *UAT Script Completion Form, Appendix VII*. During the UAT overview, the purpose and use of these two documents will be explained to the UAT Team. The testers will use the *UAT Incident/ Identification Form* to document UAT incidents. Testers must complete specific fields and provide standard information on this form. This information will be necessary to

assist the Application Team during problem resolution. The *UAT Script Completion Forms* will be used to obtain the signature and date of the tester per scenario, by script. The UAT overview will be presented by the UAT Lead and may be a combination of text/handouts and visual aids.

In addition, the UAT Team will be provided with a packet that includes instructions and four scripts for remote testing. The UAT Team will conduct remote testing at their local offices to ensure connectivity and to ensure reasonable performance when navigating through the screens.

5.11.3 System Training

The Implementation Team will also provide the UAT Team with WDTIP system training. One objective of WDTIP system training is to ensure testers understand the system design and functionality to minimize the learning curve prior to the first week of UAT testing. An additional objective of the system training is to test the *System Functionality* portion of the **WDTIP Training Curriculum** deliverable in preparation for WDTIP system training, which is scheduled for May 2000. Valuable feed-back can be gained from testers regarding the effectiveness of the *System Functionality* curriculum and an opportunity to conduct a "dry-run" will give the Implementation Team a chance to make adjustments in lesson plans, training delivery or section content. The system training for UAT will be an instructor-led session presented by the Implementation Team and will be a combination of hands-on system training with written training materials.

5.12 Testing and Retesting

All of the previous subsections of the UAT Plan have been developed to support the testing and retesting effort. This subsection will detail the processes that comprise the actual testing and retesting activities that will be undertaken by the UAT Team as they test the system from a user's perspective. Specifically this subsection will detail the following:

❑   Test Scenario and Script Execution

❑   Incident Tracking

❑   Daily UAT Meeting

❑   Script Tracking

❑   Incident Resolution

❑   Regression Testing

❑   Retesting

❑   Script Sign-off

5.12.1 Test Scenario and Script Execution

One week after completing UAT script validation, the UAT overview and WDTIP system training, the UAT Team will return to Sacramento for two weeks to begin the testing process. This testing process is based on the execution of UAT scripts created to

meet the UAT objectives. Upon arrival, the Team will be directed to the HHSDC Training Center (see the *5.7 Facilities and Equipment* subsection) where each Team member will receive a testing packet that will include:

- ❑ Three individual logons to the WDTIP system UAT region
- ❑ Set of UAT scripts
- ❑ List of individuals/CINs to use for each script
- ❑ Copy of WDTIP reference tables
- ❑ *UAT Incident/Issue Identification Forms*
- ❑ *UAT Script Completion Forms*

UAT testing will occur at the testing facility between 8:00 a.m. and 2:45 p.m. each day. At 3:00 p.m. the UAT Team will participate in a daily UAT status meeting that should adjourn no later than 5:00 p.m. Please see the *5.12.3 Daily UAT Meeting* subsection below for more information. UAT Team lunch breaks will be scheduled around batch processes and designated clock advancements whenever possible.

As mentioned above, the UAT Team will conduct remote testing prior to UAT testing at their local sites to ensure connectivity and performance of navigation through the screens. Four scripts were developed specifically for remote testing.

Time and resource permitting, each member of the UAT Team will test all scripts within the UAT scenario set. Script testing priority and order will be determined once the UAT Team has validated the scenario set. Please see *Appendix VIII* (of this document) for the *UAT Script Testing Schedule.* The UAT Team will be given subsections of the complete script set at scheduled intervals to make the testing process more manageable. The subsets of scripts will be sectioned around the scheduled batch/online updates and clock advancements built into the scenario set. Please see the *5.5 Scenarios and Scripts* subsection for more detail.

After reviewing the scripts, the testers will determine which logon and which individual or CIN (from their testing packet) should be used for the testing of each script. After logging in to the WDTIP Acceptance Test region, the tester will begin executing individual scripts. Each tester will follow the detailed step-by-step instructions included within each script and compare the expected results of each step in the script against the actual results occurring on the screen.

Those scripts where the actual results equal the expected results will be considered to have "passed" the UAT process and the tester will sign off on the script. Those scripts where the actual results of one or more steps do not match the expected results will be considered to have "failed." The testers will begin documenting incidents on the *UAT Incident/Issue Identification Form*. Please see the *5.12.2 Incident Tracking* subsection for more information. Those scripts where the actual results equal the expected results, but it is determined that a policy clarification is required or a system enhancement is requested, will be considered to be in a "pending" status. Scripts in a pending status are

considered to have passed the UAT process as they have demonstrated that the system performed as detailed in the design requirements and the tester will be expected to sign-off on the script.

All incidents discovered during the UAT process will be documented using the *UAT Incident/Issue Identification Form*. Identified incidents can have various impacts on the script execution process. These potential impacts and related processes are shown in the table below.

Table 5-10: Incident Impact Processes

If	Then	Comments
The incident allows the tester to complete the script with actual results matching the scripts expected results.	The incident should be documented using the *UAT Incident/Issue Identification Form*. The tester should continue testing the assigned scripts. The tester will provide clarification and/or validation of the incident for the UAT Lead as necessary.	The pass/fail status of the script will be dependent on the type of incident and the established UAT acceptance criteria.
The incident allows the tester to complete the script but the actual results do not match the expected results, and there are no future scripts dependent upon these expected results.	The incident should be documented using the *UAT Incident/Issue Identification Form*. The tester will complete the script (logging any incidents not related to the initial incident). The tester will provide clarification and/or validation of the incident for the UAT Lead as necessary. Move on to the next script.	The pass/fail status of the script will be dependent on the type of incident and the established UAT acceptance criteria.

If	*Then*	*Comments*
The incident allows the tester to complete the script but the actual results do not match the expected results, and there are future scripts dependent upon these expected results.	The incident should be documented using the *UAT Incident/Issue Identification Form*. Complete the script (logging any incidents not related to the initial incident). The tester will provide clarification and/or validation of the incident for the UAT Lead as necessary. The tester will continue testing scripts until he/she reaches a script that cannot be tested because it is dependant on the results of the script with the recorded incident. The dependent script will be passed over until the incident in the initial script is resolved and the script is retested so the dependent script can be adequately tested. The dependent script will be tested once the incident in the initial script has been resolved.	The UAT Lead will keep track of all incidents affecting future scripts and will notify the UAT Team when they are approaching an affected script. Please see the *5.12.2, Incident Tracking* subsection. The UAT Lead will keep track of when an incident has been fixed and the script is ready to be retested.
The incident does not allow the tester to complete a script and there are no future scripts dependent upon expected results.	The incident should be documented using the *UAT Incident/Issue Identification Form*. The tester will provide clarification and/or validation of the incident for the UAT Lead as necessary. The tester should move on to the next script. The script will be retested when the UAT Lead advises the tester that the incident has been fixed.	The UAT Lead will keep track of when an incident has been fixed and the script is ready to be retested.

If	Then	Comments
The incident does not allow the tester to complete a script and there are future scripts dependent upon expected results.	The incident should be documented using the *UAT Incident/Issue Identification Form*. The tester will provide clarification and/or validation of the incident for the UAT Lead as necessary. The tester should move on to the next script. The tester will continue testing scripts until he/she reaches a script that cannot be tested because it is dependant on the results of the script with the recorded incident. The dependent script will be passed over until the incident in the initial script is resolved and the script is retested so the dependent script can be adequately tested. The dependent script will be tested once the incident in the initial script has been resolved.	The UAT Lead will keep track of all incidents affecting future scripts and will notify the UAT Team when they are approaching an affected script. Please see the *5.12.2, Incident Tracking* subsection. The UAT Lead will keep track of when an incident has been fixed and the script is ready to be retested.

5.12.2 Incident Tracking

During the execution of UAT scenarios, the UAT Team will identify incidents (within Project scope) that may prevent them from signing-off on a particular script. It will be necessary to track the daily overall pass/fail/pending (see the *5.12.4 Script Tracking* subsection for definitions) status of UAT scripts as well as any identified incidents or issues associated with any non-passing UAT scripts. This subsection will document the processes and mechanisms for tracking UAT incidents.

If a UAT Team member identifies an incident while executing a test scenario, that person will first confirm the incident's validity with the UAT Lead and/or any available UAT support staff from the WDTIP Implementation or Application Teams. This check will help minimize the submission of incidents that arise as a result of user error or that are outside of the scope of the Project. If the incident is determined to be valid or a question remains regarding the incident's validity, the UAT Team member will document the incident on a *UAT Incident/Issue Identification Form*. Instructions for completing the form will be covered during the UAT overview. Once the incident has been recorded on the *UAT Incident/Issue Identification Form,* the UAT Team member will give it to the UAT Lead or UAT support staff for entry into the WDTIP PTS.

The PTS is an Access database that is used to record and monitor both incidents and issues that arise during the course of WDTIP. An incident, as defined for the purposes of

UAT, occurs when the actual result of one or more steps in a UAT script differs from the script's expected results or when system functionality differs from user expectations. The primary purpose of the PTS is to help monitor the progression of incidents through the Change Control Process and produce reports that provide information pertaining to such items as status and assignment of incidents. Once an incident has been entered into the PTS at the testing facility, the *UAT Incident/Issue Identification Form* will be returned to the UAT Team member who identified the incident. The Team member will be instructed to keep this form to aid discussion in the daily UAT meeting (see the *5.12.3 Daily UAT Meeting* subsection below).

5.12.3 Daily UAT Meeting

All incidents identified during the testing day will be discussed during the daily UAT meeting. Every afternoon, starting at approximately 3:00 p.m., the UAT Team will meet with the UAT Lead, and representatives from the Implementation and Application Teams, to:

- ❑ Clarify WDTIP functionality questions among the UAT Team
- ❑ Communicate and validate new incidents and effective testing approaches
- ❑ Eliminate redundant incidents
- ❑ Compile daily results
- ❑ Discuss daily UAT status
- ❑ Update status on outstanding incidents
- ❑ Present plans for the next day (code migrations, script schedule, and retesting)

The UAT Lead will generate and distribute daily reports from the PTS and the STS for the daily UAT meeting (see the *5.12.4 Script Tracking* subsection). These reports will provide a status of UAT activities including incident status and a comprehensive script status (pass, fail or pending).

Application Team representation at the daily UAT meeting is necessary to help validate and appropriately assign any incidents discovered during the day. Having an Application Team representative available is also beneficial because he/she will be able to hear about any identified incidents first hand from the UAT Team. This first hand information will allow the Application Team to better grasp UAT incidents and begin the incident resolution process immediately (see the *5.12.5 Incident Resolution* subsection).

At the conclusion of the daily UAT meeting, incidents and scripts requiring updates in status or description will be made in the PTS and the STS (see the *5.12.4 Script Tracking* subsection) by the appropriate Team member. The UAT Lead will track daily and overall incident and scenario status for documentation in the **User Acceptance Test Sign-off** deliverable. This status information will also be presented in the Project's morning status meeting (see *5.13 UAT Results* subsection).

5.12.4 Script Tracking

It will be necessary to track the status of the scripts within the UAT scenario set to effectively manage the testing process as well as monitor and report testing progress. After the daily UAT meeting, the UAT Lead will compile the status of all UAT scripts. Any changes to a script's status will be entered into the STS.

This tracking tool is an Access database designed to track the status of scenarios. The system contains the UAT scripts along with their expected results. Maintenance of test scripts also occurs within the tool. Test results are recorded to identify which scripts have passed and which have not. The UAT Lead will be able to attach an overall status of pass, fail or pending to any script. These classifications are defined as follows:

Table 5-11: Script Status

Status	Definition
Pass	Scripts that have been tested by each member of the UAT Team and actual results equal expected results.
Fail	Scripts that have not been signed-off on by one or more members of the UAT Team because actual results do not equal expected results.
Pending	A script that has not been classified as pass or fail because the Team is waiting for an action to occur before the script can be fully tested (e.g., the Team is awaiting specific MEDS logons).

For any non-passing scripts, the UAT Lead will attach a description of the incident or policy issue to a specific step within the script. Once script status has been updated in the STS, a report can be generated that shows all issues affecting a particular script.

5.12.5 Incident Resolution

Incidents will be validated during the daily UAT meeting. During the meeting, the participants will collectively decide if an incident is an "enhancement" or a "fix." An enhancement (also referred to as an issue) is any incident that is outside of the Project's scope or needs clarification to determine if it is within Project scope. Any incident that is determined to be an enhancement will be entered into the issue tracking log and will be subject to the established protocol of the *Configuration Control Process* as outlined in the **Configuration Management Plan**. If an incident is identified as an enhancement, the STS will be updated by the UAT Lead to show the script status as pass. For purposes of UAT, any script that has an incident determined to be an enhancement will be considered to have passed the UAT process.

Fixes are identified as incidents within the scope of the Project and, therefore, do not need to be resolved through the *Configuration Control Process*. They will be reported and tracked in accordance with the approved **Configuration Management Plan**. Once an incident has been validated at the daily UAT meeting, it's status will be updated in the

STS (if necessary) and PTS and assigned to the appropriate WDTIP Team Lead for resolution. The Team Lead will then reassign the incident to a programmer to make the necessary adjustments to the affected program(s). When the incident has been resolved the Team Lead, Tech Support Lead, and UAT Lead will meet to discuss migration of the change into the System Test environment for integration testing. Subsequently, it will be migrated to the UAT environment for regression testing. Regression testing (see the *5.12.6 Regression Testing* subsection for more information) will be executed on any impacted scripts before it is released to the UAT Team for retesting (see the *5.12.7 Retesting* subsection).

Following the successful migration of a fix to the UAT environment the STS will be updated by the UAT Lead to reflect any action taken on an existing incident.

5.12.6 Regression Testing

Regression testing will be incorporated into the UAT activities to ensure that program code migrations to the UAT environment do not adversely affect previously stable areas of the WDTIP system. The regression test must be a repeatable process, meaning that a regression test held one time is basically the same as one held the following time, where no unplanned variations in the test process exist. Regression testing will occur following each new code migration to the UAT environment where a full retest of all scripts is not planned. Performed properly, regression testing provides proof of correctness from one build to another.

Prior to the migration of any WDTIP system code changes to the UAT region, a snapshot of the existing data and code will be taken as a precautionary measure in case the migration adversely affects previously stable areas of the WDTIP system. If adverse changes do occur, the snapshot can be used to refresh the original system environment until appropriate fixes can be made.

Identified scripts will then be retested using back-up sets of CINs that contain data identical to the CINs used in the original testing effort (see the *5.9 Test Data* subsection). These regression CINs will be isolated from the CINs being used during ongoing UAT activities to ensure data entegrity. This will be accomplished by applying only necessary online, batch and time clock logic to the identified regression testing CINs within the UAT environment.

To determine which scripts will be tested to ensure a successful code migration to the UAT environment, it will be necessary to determine what functionality is affected by the program changes. The regression test script selection process is depicted in the following diagram (see *Figure 5-2*). After a migration to the UAT region, the Implementation Team will consult with the Application Team to determine what code changes have been made and what parts of the system functionality these code changes will affect. Once any impacted functions (Functionality C in *Figure 5-2*) of the WDTIP system have been identified, UAT scripts that address the impacted functionality will be identified (Script 1 and Script 4 in *Figure 5-2*). In most cases, it will not be necessary to perform regression testing on all of these identified scripts due to the inclusion of various permutations around the same functionality within the UAT scenario set. Specific scripts to be used for

regression testing will be identified from the set of affected scripts based on the scope of
the migrated changes.

Figure 5-2 – Regression Test Script Selection
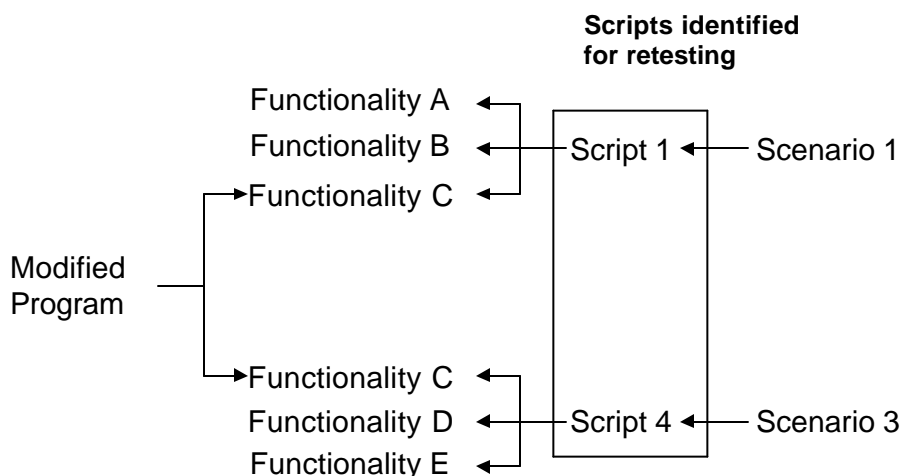
5.12.7 Retesting

Time permitting, the UAT Team will retest non-passing scripts once incident fixes have
been migrated into the UAT environment. UAT incidents will receive high priority in the
incident resolution process so they can be resolved and retested while the UAT Team is
still in Sacramento. To retest any fixed incidents, it will be necessary to use a back-up
CIN with data identical to that of the CIN used when the original incident was created.
These back-up CINs with appropriate data will be created during the data creation
process (see the *5.9 Test Data* subsection). Back-up CINs are required to retest because
using the original CIN will most likely require that the clock used for ongoing UAT
testing be set back (due to the progression of testing and the time to complete the fix),
which would impact the ongoing testing of other UAT scripts. Once a fix has been made,
applicable back up CINs will be given to all UAT Team members. All Team members
will then retest the scenario associated with the fixed incident. If clock advancements or
batch processes need to be applied to achieve expected results, the Application Team will
isolate the retest CINs and apply the necessary program logic to only retest CINs so
ongoing UAT efforts will not be impacted.

If the fix resolves the incident, the UAT Lead will update the PTS and the STS. The UAT
Team members that identified the incident will then sign-off on the script if no other
incidents arise during the retest. If the incident is not fixed or another incident is
identified during the retesting effort, the UAT Lead will re-open and reassign the
incident, or create a new one in the PTS as well as update the actual results section of the
STS.

5.12.8 Script Sign-off

When testers are able to complete a script without identifying any incidents (within Project scope), they will sign and date the individual script sign-off sheet provided to them in their testing packet next to the associated scenario and script. This signature will signify the testers' acceptance of the WDTIP functionality contained within the tested script. It is assumed that all UAT Team members will have the authority to represent their county/consortia system when signing-off on successfully completed UAT scripts. These sign-off sheets will be applied against UAT Acceptance Criteria (see the *5.10 Acceptance Criteria* subsection) and compiled for the **User Acceptance Test Sign-off** deliverable.

5.13 UAT Results

UAT results will be compiled and detailed in the **User Acceptance Test Sign-off** deliverable. These results will include daily summaries of testing activities and a final report of UAT findings. These results will be shown in table or graphic format wherever possible.

The UAT results subsection includes a description of the reports that will be generated on a daily basis during the UAT testing/retesting activities and a description of the summary reports and graphs that will be generated upon completion of the UAT activities. Also included is an overview of the UAT sign-off process.

5.13.1 Daily UAT Status

A daily status of UAT activities will be provided each day during the testing and retesting activities. This report will be generated utilizing the STS and will identify the status of scenarios and their associated scripts. The information will be reported on a cumulative basis and includes:

- ❑ Total Number of Scripts
- ❑ Number of Scripts that Passed by Scenario Area
- ❑ Number of Scripts that Failed by Scenario Area
- ❑ Number of Scripts that Pending by Scenario Area
- ❑ Number of Scripts Being Retested by Scenario Area
- ❑ Number of Scripts In Progress by Scenario Area
- ❑ Number of Invalid Scripts (if any) by Scenario Area
- ❑ Total Number of Scripts Not Started by Scenario Area

The information provided will include a narrative summary of failed scripts including the reason for the failures, incidents that occurred and issues that were discovered. A *Sample Daily UAT Status Report* is included as *Appendix IX.*

5.13.2 Daily Incident Totals

A daily status of incidents added to the STS database will be provided each day during the UAT testing and retesting activities. The information will be reported in both graph

and table formats and will be reported on a cumulative basis. The incident information reported on will be classified by priority and includes the following:

- Number of Incidents Open
- Number of Incidents with Work in Progress
- Number of Incidents with Incident Testing Completed
- Number of Incidents Ready for Migration
- Number of Incidents Ready for Retesting
- Number of Incidents Closed
- Number of Incidents Cancelled
- Number of Total Incidents

A *Sample Daily UAT Incident Status Report* is included as *Appendix X*.

5.13.3 Daily Program Change Request Totals

A daily status of Program Change Requests (PCRs) will be provided each day during the UAT testing/retesting activities. The information will be reported in both graph and table formats and will be reported on a cumulative basis. The following information will be reported classified by scenario area and priority:

- Number of PCRs Not Started
- Number of PCRs in Progress
- Number of PCRs with Programming Completed
- Number of PCRs with Peer Review Requested
- Number of PCRs with Peer Review Completed
- Total Number of PCRs
- Total PCRs with Associated Task Hours
- Total Hours Estimated for Completion of Outstanding PCRs

A *Sample Daily UAT PCR Status Report* is included as *Appendix XI.*

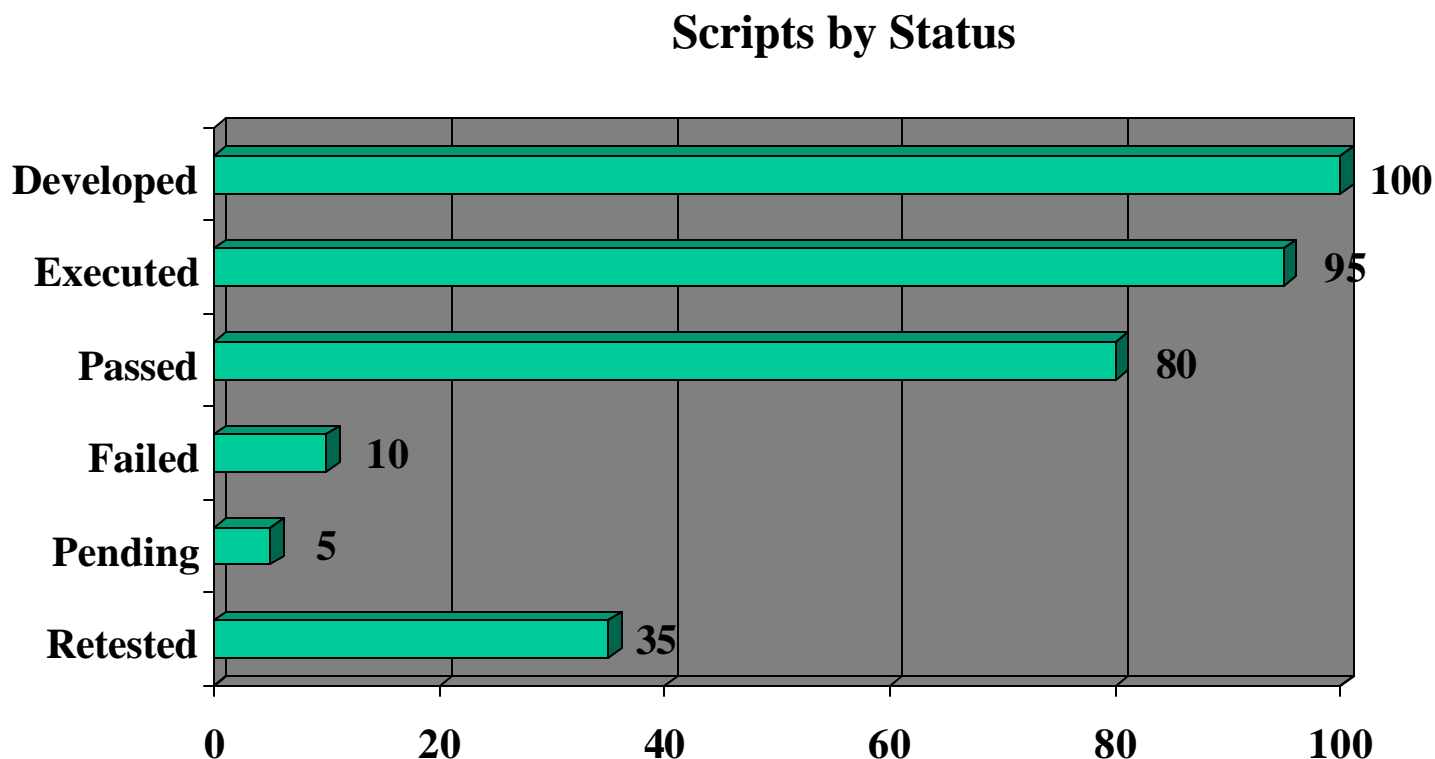### 5.13.4 Summary Testing Findings

At the completion of UAT activities, results of the testing will be summarized and reported as part of the **User Acceptance Test Sign-off** deliverable. This report will include a narrative summary of the UAT results, including a list of participants and their respective county or agency, the percentage of scripts passing and failing, outstanding incidents or issues and a summary of the reasons for script failures, and an action plan to fix/resolve all outstanding incidents/issues.

The report will also provide summarized information displayed graphically. These reports are:

- ❏ Number of Scripts by Status (Developed, Executed, Passed, Failed, Pending, Retested)
- ❏ Number of Scripts by Priority and Status (High, Medium, Low)
- ❏ Number of Scripts by Scenario Area
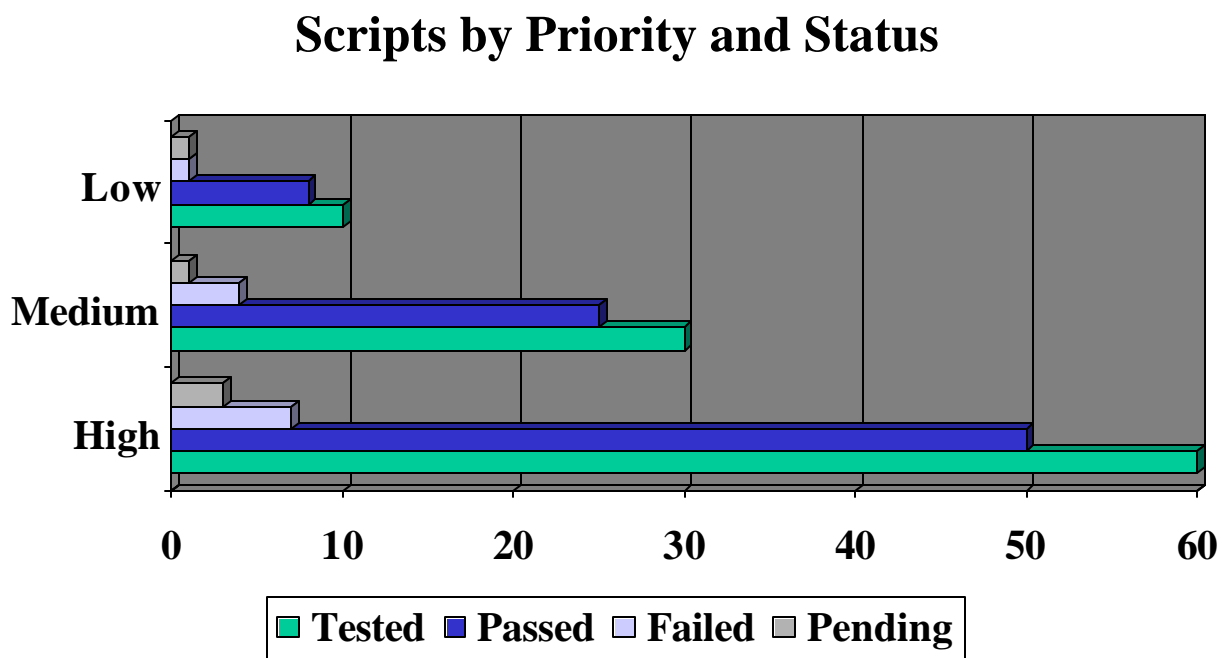- ❏ Number of Incidents by Scenario Area

5.13.4.1 Number of Scripts by Status

This graph will display the number of scripts developed, executed, passed, failed, pending, and the number of scripts that need to be retested. A sample graph is presented below.

Scripts by Status

Status	Value
Developed	100
Executed	95
Passed	80
Failed	10
Pending	5
Retested	35

(Horizontal bar chart, x-axis from 0 to 100 in increments of 20)

5.13.4.2 Number of Scripts by Priority and Status

This graph will display the number of scripts executed during UAT, number of scripts
that passed, number of scripts that failed, and number of scripts in a pending status by the
priority assigned to the script. A sample graph is presented below.

Scripts by Priority and Status

Legend: ■ Tested ■ Passed □ Failed ■ Pending

5.13.4.3 Number of Scripts by Scenario Area

This graph will display the number of scripts completed, number of scripts that passed, number of scripts that failed, and number of scripts pending by scenario area. These include areas such as Navigation, Security, Time Clocks, Conversion, File Extract Loads, Non-California Program Participation, Diversion, Child Support Reimbursement, Supportive Services and Reports. A sample graph is presented below.

Scripts by Scenario Area
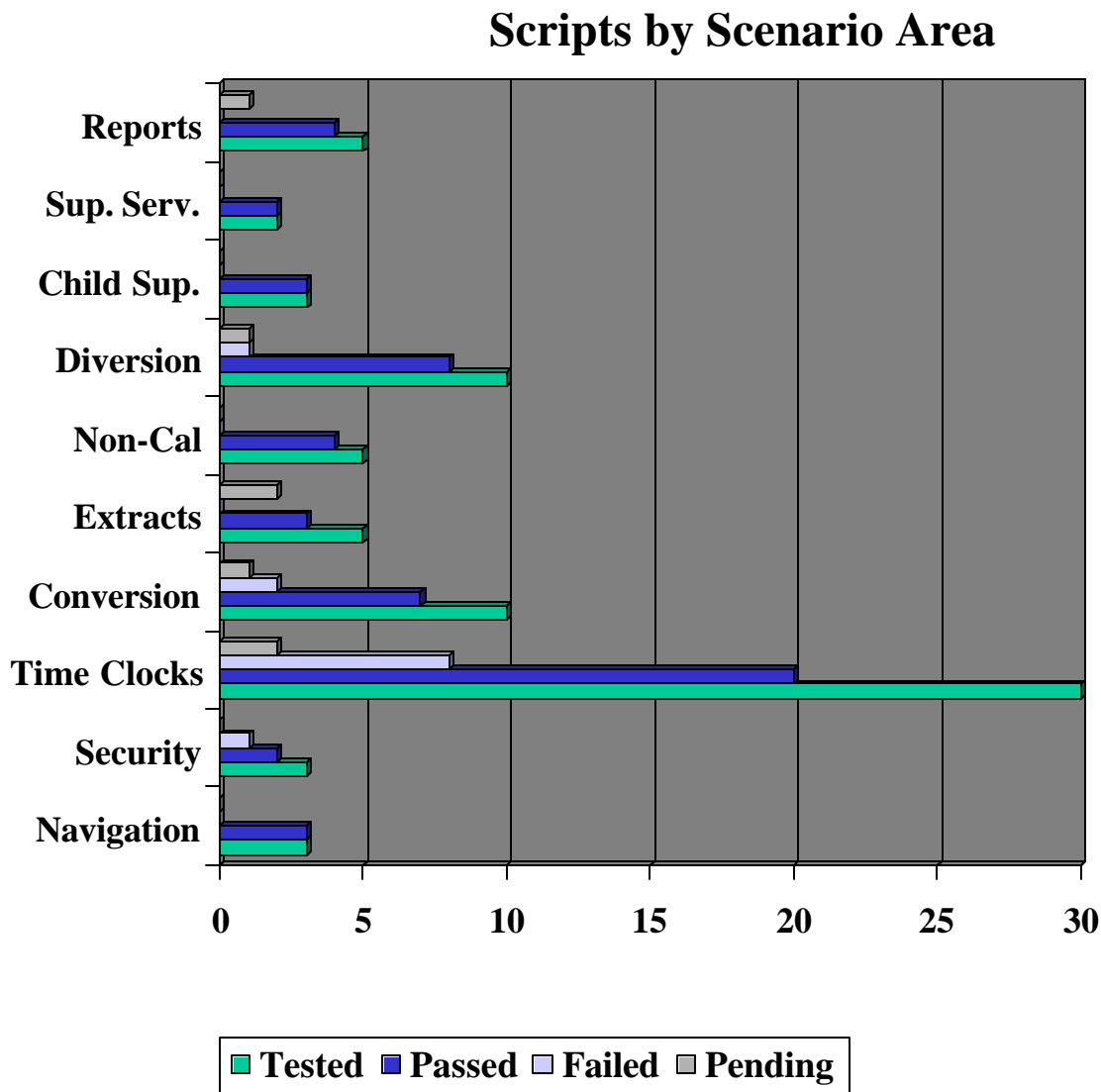
5.13.4.4 Number of Incidents by Scenario Area
This report will display the number of incidents logged into the tracking database by scenario area. A sample report is printed below.

Incidents by Scenario Area

Legend: ■ Incidents Logged ■ Unresolved Incidents
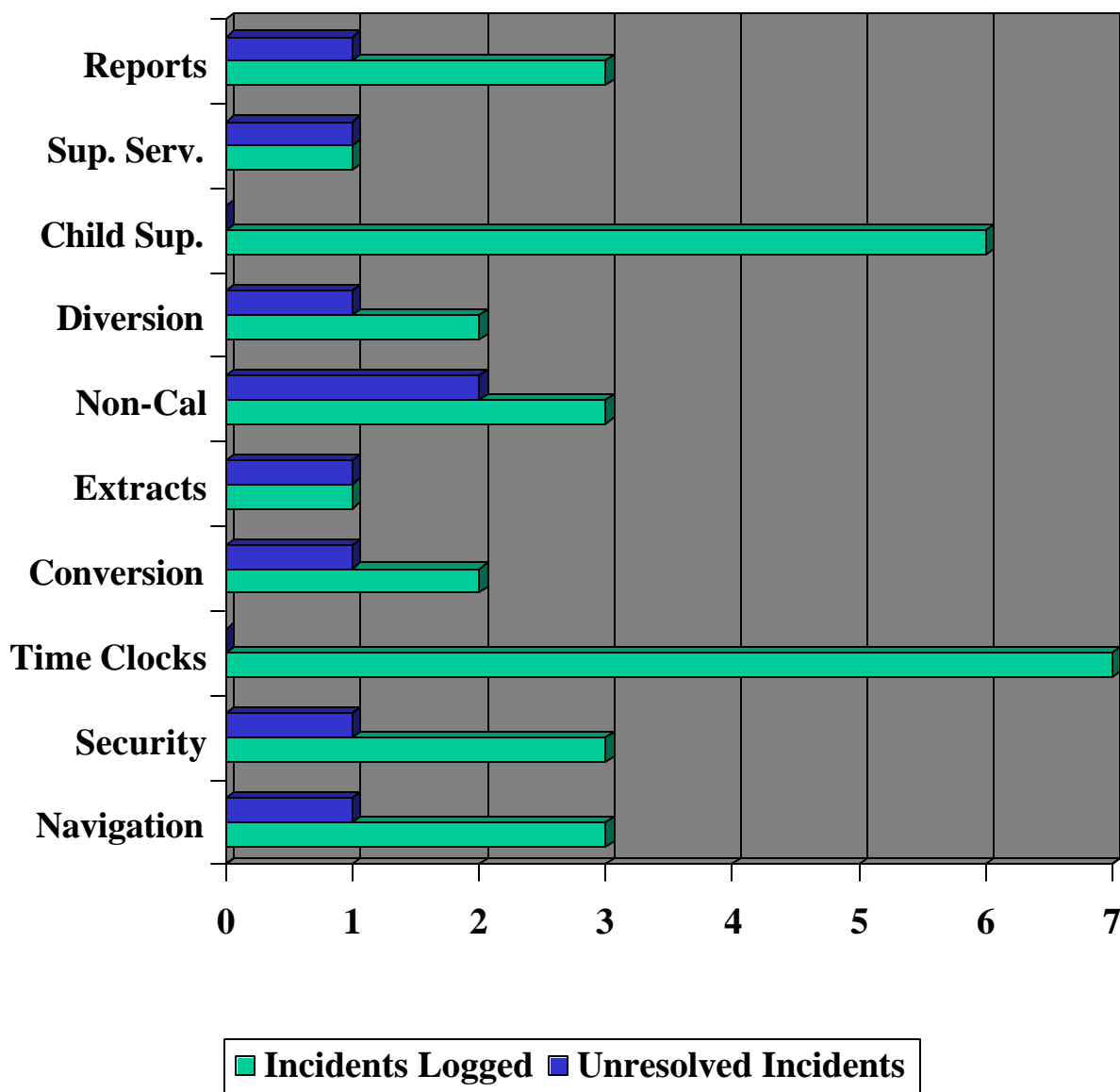
5.13.5 User Acceptance Test Sign-off

The UAT sign-off is a formal process whereby the UAT Team members certify that the WDTIP system requirements have been successfully met by executing and signing off on specific test scripts. CDSS subsequently will make a decision that UAT is completed based upon evaluating the UAT results against the acceptance criteria. Acceptance criteria have been developed by CDSS in conjunction with HHSDC. Please see *5.10 Acceptance Criteria* subsection for more information.

UAT is successfully concluded when the UAT Team has passed all scripts or when failed scripts have been resolved (i.e., tied to CDSS policy issues or incidents that will be fixed with future releases of the application).